

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2016

Bc. Marek Kříž



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY

A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

DEPARTMENT OF BIOMEDICAL ENGINEERING

**SROVNÁNÍ OPTIMALIZAČNÍCH METOD PRO ODHAD
PERFÚZNÍCH PARAMETRŮ**

COMPARISON OF OPTIMIZATION METHODS FOR PERFUSION PARAMETERS ESTIMATION

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Marek Kříž

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Martin Mézl

BRNO 2016



Diplomová práce

magisterský navazující studijní obor **Biomedicínské a ekologické inženýrství**

Ústav biomedicínského inženýrství

Student: Bc. Marek Kříž

ID: 147500

Ročník: 2

Akademický rok: 2015/16

NÁZEV TÉMATU:

Srovnání optimalizačních metod pro odhad perfúzních parametrů

POKYNY PRO VYPRACOVÁNÍ:

1) Seznamte se a popište základní principy ultrazvukového perfúzního zobrazování. 2) Proveďte rešerši metod matematické optimalizace pro odhad perfúzních parametrů založený na proložení křivky koncentrace pomocí dilučních modelů. 3) Vybrané metody implementujte v programovém prostředí Matlab. Pro proložení dat modelem využijte funkce z optimalizačního toolboxu. Tyto funkce řádně popište. 4) Seznamte se s metodami globální optimalizace. Vybte některou ze stochastických metod globální optimalizace a implementujte ji pro řešení prokládání dat modelem. 5) Realizujte globální optimalizaci pomocí některého z evolučních algoritmů. Všechny implementace otestujte na uměle vytvořených datech a datech z ultrazvukového fantomového přípravku. 6) Proveďte srovnání optimalizačních metod vzhledem k možnosti určení globálního extrému, nastavení parametrů algoritmu a vzhledem ke správnosti odhadovaných parametrů.

DOPORUČENÁ LITERATURA:

[1] FLETCHER, R. Practical Methods of Optimization. 2nd edition. New York : Wiley, 2000. 450 s. ISBN 978--471-49463-8.

[2] STROUTHOS, C. et al. Indicator Dilution Models for the Quantification of Microvascular Blood Flow With Bolus Administration of Ultrasound Contrast Agent. IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, 2010, 57, 6, s. 1296-1310.

Termín zadání: 8.2.2016

Termín odevzdání: 20.5.2016

Vedoucí práce: Ing. Martin Mézl

Konzultant diplomové práce:

prof. Ing. Ivo Provazník, Ph.D., předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Náplní této diplomové práce je pochopení principu ultrazvukového zobrazení, matematických modelů a využití pro odhad perfúzních parametrů z koncentračních křivek. Práce se zabývá globálními optimalizačními algoritmy pro nalezení parametrů aproximace skutečných dat modelovými křivkami. Obsahuje také srovnání jednotlivých metod a použitých funkcí.

KLÍČOVÁ SLOVA

Ultrazvuk, perfúzní zobrazení, matematický model, globální optimalizace, netopýří algoritmus, genetický algoritmus, simulované žíhání

ABSTRACT

The content of this thesis is to understand the principle of ultrasound imaging and mathematical models used to estimate perfusion parameters of concentration curves. Thesis deals with global optimization algorithms for finding parameters, an approximation of the actual data model curves. It also includes a comparison of different methods and used functions.

KEYWORDS

Ultrasound, perfusion image, mathematical model, global optimization, bat algorithm, genetic algorithm, simulated annealing

KŘÍŽ, Marek *Srovnání optimalizačních metod pro odhad perfúzních parametrů*: diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav biomedicínského inženýrství, 2015/2016. 82 s. Vedoucí práce byl Ing. Martin Mézl

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Srovnání optimalizačních metod pro odhad perfúzních parametrů“ jsem vypracoval(a) samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor(ka) uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil(a) autorská práva třetích osob, zejména jsem nezasáhl(a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(a) následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora(-ky)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Martinu Mězlovi za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno

.....

podpis autora(-ky)

OBSAH

Úvod	11
1 Teoretický úvod	12
1.1 Ultrazvuk	12
1.1.1 Vznik ultrazvukového signálu	13
1.1.2 Ultrazvukové sondy	13
1.1.3 Zobrazení	14
1.1.4 Ultrazvukové módy	15
1.1.5 Kontrastní látky	16
1.1.6 Perfuzní zobrazení	17
2 Matematické modely	19
2.1 Matematické modely	19
2.1.1 Klasifikace matematických modelů	19
2.1.2 Farmakokinetické modely	21
2.2 Základní farmakokinetické parametry	21
2.2.1 Jednokompartmentové farmakokinetické modely	24
2.2.2 Vícekompartmentové farmakokinetické modely	24
2.3 Matematické modely pro perfuzní zobrazení	26
2.3.1 Log-normální rozložení	26
2.3.2 Erlang rozložení	27
2.3.3 Gamma rozložení	28
2.3.4 LDRW	29
3 Optimalizace pro modelová data	30
3.1 Modelování šumu	30
3.2 Popis funkcí	31
3.3 Výstupy	32
4 Globální optimalizace	34
4.1 Simulované žíhání	35
4.2 Genetické algoritmy	35
4.2.1 Generování počáteční populace	36
4.2.2 Genetické operátory	36
4.2.3 Netopýří algoritmus	38

5 Implementace metod globální optimalizace	40
5.1 Vstupní parametry	40
5.2 Popis uživatelského rozhraní	41
5.3 Funkce	42
6 Testování	46
6.1 Použitá data	46
6.2 Výstupy	49
6.2.1 Výstupy pro data1	49
6.2.2 Výstupy pro data3	56
6.2.3 Výstupy pro data2	62
6.2.4 Výstupy pro data4	62
7 Diskuze	63
8 Závěr	64
Literatura	65
Seznam symbolů, veličin a zkratk	69
Seznam příloh	70
A Tabulka výsledků pro data1	71
B Tabulka výsledků pro data2	72
C Tabulka výsledků pro data3	73
D Tabulka výsledků pro data4	74
E Grafy výsledků pro data2	75
F Grafy výsledků pro data4	78
G Druhá příloha	81
H Obsah přiloženého CD	82

SEZNAM OBRÁZKŮ

1.1	Ukázka druhů ultrazvukových sond	14
2.1	Plocha pod křivkou AUC, převzato z [18]	23
2.2	Jednokompartmentový model distribuce léčiva	24
2.3	Vícekompartmentový model s rozdělením na krevní oběh a tkáň	25
2.4	Průběh log-normální distribuce	27
2.5	Průběh gamma distribuce	28
2.6	Průběh LDRW distribuce	29
3.1	Průběh log-normální distribuce se šumem a původní signál	30
3.2	Průběh log-normální distribuce se šumem a nově generovaný signál	31
3.3	Srovnání jednotlivých funkcí pro log-normální vstup	32
5.1	Uživatelské rozhraní GUI	41
5.2	Vývojový diagram simulovaného žíhání	43
5.3	Vývojový diagram genetického algoritmu	44
5.4	Vývojový diagram netopýřího algoritmu	45
6.1	Schématické zapojení fantomu	46
6.2	Srovnání pomalých a rychlých dat	47
6.3	Aproximace dat pomocí netopýřího algoritmu	50
6.4	Aproximace dat pomocí genetického algoritmu	50
6.5	Aproximace dat pomocí simulovaného žíhání	51
6.6	Srovnání funkčních hodnot pro změnu velikosti populace	51
6.7	Srovnání funkčních hodnot pro změnu počtu iterací	52
6.8	Srovnání funkčních hodnot při změně výpočetního algoritmu	52
6.9	Pozice parametrů pro LDRW funkci, velikost populace [10]	54
6.10	Pozice parametrů pro LDRW funkci, velikost populace [20]	54
6.11	Pozice parametrů pro LDRW funkci, velikost populace [50]	55
6.12	Pozice parametrů pro LDRW funkci, velikost populace [100]	55
6.13	Aproximace dat pomocí netopýřího algoritmu	56
6.14	Aproximace dat pomocí genetického algoritmu	57
6.15	Aproximace dat pomocí simulovaného žíhání	57
6.16	Srovnání funkčních hodnot pro změnu velikosti populace	58
6.17	Srovnání funkčních hodnot pro změnu počtu iterací	58
6.18	Srovnání funkčních hodnot při změně výpočetního algoritmu	59
6.19	Pozice parametrů pro log-normální funkci, velikost populace [10]	60
6.20	Pozice parametrů pro log-normální funkci, velikost populace [20]	60
6.21	Pozice parametrů pro log-normální funkci, velikost populace [50]	61
6.22	Pozice parametrů pro log-normální funkci, velikost populace [100]	61
E.1	Aproximace dat pomocí netopýřího algoritmu	75

E.2	Aproximace dat pomocí genetického algoritmu	75
E.3	Aproximace dat pomocí simulovaného žíhání	76
E.4	Srovnání výstupů pro změnu populace	76
E.5	Srovnání výstupů pro změnu iterace	76
E.6	Pozice parametrů pro netopýří algoritmu, populace n=10	77
E.7	Pozice parametrů pro netopýří algoritmu, populace n=20	77
E.8	Pozice parametrů pro netopýří algoritmu, populace n=50	77
E.9	Pozice parametrů pro netopýří algoritmu, populace n=100	77
F.1	Aproximace dat pomocí netopýřího algoritmu	78
F.2	Aproximace dat pomocí genetického algoritmu	78
F.3	Aproximace dat pomocí simulovaného žíhání	79
F.4	Srovnání výstupů pro změnu populace	79
F.5	Srovnání výstupů pro změnu iterace	79
F.6	Pozice parametrů pro SA, populace n=10	80
F.7	Pozice parametrů pro SA, populace n=20	80
F.8	Pozice parametrů pro SA, populace n=50	80
F.9	Pozice parametrů pro SA, populace n=100	80

SEZNAM TABULEK

1.1	Frekvence ultrazvuku v diagnostice [25]	15
3.1	Srovnání koeficientu pro jednotlivé funkce	32
3.2	Srovnání Pearsonova korelačního koeficientu pro jednotlivé funkce	33
6.1	Tabulka testovaných dat	47
6.2	Výpočetní náročnost algoritmu	48
6.3	Nejmenší funkční hodnoty pro jednotlivé kombinace	49
6.4	Nejmenší funkční hodnoty pro jednotlivé kombinace	56
6.5	Nejmenší funkční hodnoty pro jednotlivé kombinace	62
6.6	Nejmenší funkční hodnoty pro jednotlivé kombinace	62
A.1	Tabulka hodnot pro data exp13_AIF_inp_tis_111007.mat	71
B.1	Tabulka hodnot pro data exp14_roimale_inp_tis_110907.mat	72
C.1	Tabulka hodnot pro data exp17_roivelke_inp_tis_111007.mat	73
D.1	Tabulka hodnot pro data exp15_AIF_inp_tis_111007.mat	74
G.1	Použité testovací nastavení	81
G.2	Technické parametry testovacího počítače	81
H.1	Seznam příložených souborů na CD	82

ÚVOD

Tato práce se zabývá ultrazvukovým perfuzním zobrazením. Je zde rozebrán teoretický úvod ultrazvuku, vznik signálu, používané ultrazvukové sondy a kontrastní látky pro perfuzní zobrazení. Při použití ultrazvukové perfúzní analýzy je základem postup, kdy jsou analyzované perfúzní křivky získané po aplikaci kontrastní látky prokládány matematickými modely. Model by měl reálně napodobit naměřené křivky. Proto se dále zabývám metodami matematického modelování. Jsou zde rozebrány jednotlivé druhy modelů, farmakokinetické modely a modely pro perfuzní zobrazení. Dále je zde generování perfuzního signálu a jeho proložení pomocí optimalizační FIT funkce. Praktickou částí je vytvoření programů pro globální optimalizace pro aproximaci dat pomocí různých funkcí. Jsou zde rozebrány a implementovány genetický model, netopýří model a simulované žíhání. Jednotlivé modely a funkce jsou otestovány na skutečných datech a výstupy jsou uvedeny v práci.

1 TEORETICKÝ ÚVOD

1.1 Ultrazvuk

Signálem pro ultrazvuk je mechanické vlnění hmotného prostředí v rozsahu kmitočtů od 16 kHz do 1 GHz. Pro lékařské aplikace v diagnostice se používají frekvence v rozsahu 2-30 MHz. Ultrazvukové kmity se šíří formou vlnění, v měkkých tkání a tekutinách lidského těla se šíří formou podélného vlnění. V kostech se šíří formou vlnění příčného. Vlnění je popsáno vztahem 1.1:

$$\lambda = \frac{c}{f} \quad (1.1)$$

kde λ [m] je vlnová délka, f [Hz] je frekvence a c [$m \cdot s^{-1}$] je rychlost šíření ultrazvukové vlny v daném prostředí (tzv. fázová rychlost).

Ultrazvuk nachází uplatnění v mnoha oblastech diagnostiky či řízené terapie. Hlavní využití je v zobrazování abdominální oblasti, kde jsou orgány snadno přístupné. Pro šíření ultrazvuku v biologické tkáni (obecně v pevných látkách a kapalinách) je dán vztah 1.2:

$$c = \sqrt{\frac{E}{\rho}} \quad (1.2)$$

kde E [Pa] je Youngův modul pružnosti a ρ [$kg \cdot m^{-3}$] je hustota prostředí. Obecně platí, že rychlost šíření ultrazvuku v měkkých tkáních odpovídá rychlosti šíření ultrazvukové vlny ve vodě. Tato hodnota se pohybuje kolem $1500 \text{ m} \cdot \text{s}^{-1}$.

Při průchodu ultrazvukového signálu skrz různá prostředí, dojde na jejich rozhraní k lomu svazku, který popisuje vztah 1.3 :

$$\frac{\sin(\Theta_1)}{\sin(\Theta_2)} = \frac{c_1}{c_2} \quad (1.3)$$

kde Θ_1 je úhel dopadu ultrazvukového signálu, Θ_2 je úhel lomu signálu a c_1 a c_2 jsou rychlosti šíření ultrazvuku v jednotlivých prostředích.

Biologické účinky ultrazvuku jsou způsobeny absorpcí zvukové energie ve tkáni a jsou závislé na intenzitě, frekvenci a době expozice. Mezi tyto účinky patří tepelné účinky, vznikající třením kmitajících částic. Tyto účinky jsou největší na rozhraní dvou tkání s různou akustickou impedancí. Dále jsou to fyzikálně chemické a disperzní účinky, které mohou způsobit excitaci molekul a tím rychlejší chemické reakce. Ultrazvukem je možné připravit jemné suspenze, emulze, pěny či aerosoly. Jako další se uvádějí chemické a elektrochemické účinky, při kterých může dojít k depolymerizaci vysokomolekulárních látek a také mohou vznikat volné radikály ve vodném prostředí.

Mechanické účinky ultrazvuku popisuje veličina zvaná mechanický index (MI), která udává možnost vzniku kavitací v tekutině. Mechanický index je definován jako:

$$MI = \frac{P_{neg}}{\sqrt{f}} \quad (1.4)$$

kde P_{neg} je maximum negativního tlaku a f je frekvence ultrazvuku. Mechanický účinek se nejvíc projeví v oblasti fokusovaného svazku. Na okrajích může být 2-3 krát menší než je jeho maximum a klesá s rostoucí hloubkou. Mechanické účinky se považují za minimální, pokud hodnota mechanického indexu je menší nebo rovna 1. [23] [17].

1.1.1 Vznik ultrazvukového signálu

Ultrazvuk lze generovat pomocí magnetostrikčních generátorů nebo piezoelektrickými generátory. Magnetostrikční generátor vyvolá ultrazvukové kmity v prostředí kolem magnetické tyčinky, která je umístěna ve střídavém magnetickém poli. Využívá se zde feromagnetického jevu, tj. změny objemu feromagnetických látek vlivem okolního magnetického pole. Tyto generátory jsou schopny generovat frekvence do $60kHz$ ale s vysokým výkonem.

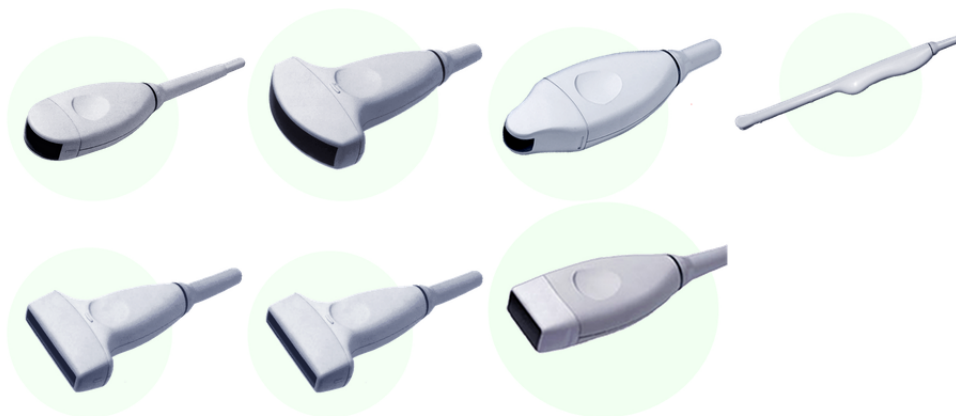
Piezoelektrické generátory pracují na principu nepřímého piezoelektrického jevu, tj. na deformaci některých materiálů vlivem elektrického napětí. Pro lékařské aplikace se používá výbrus krystalu křemene. Přivede-li se střídavé napětí o rezonančním kmitočtu destičky na její opačné konce, začne destička konat pružné kmity. Tyto kmity se následně šíří okolním prostředím. Aby nedocházelo k útlumu, je kmitající destička uložena v kapalině (oleji). [23] [16]

1.1.2 Ultrazvukové sondy

V lékařství se užívá několik druhů ultrazvukových sond. Podle geometrického tvaru rozdělujeme sondy na sektorové, konvexní a lineární. Snaha o získání detailnějších obrazů orgánů ležících v blízkosti tělesných dutin vedla ke konstrukci endokavitárních sond. Jsou to trans-vaginální sondy, trans-rektální sonda, transezofageální sonda, per-operační a laparoskopické sondy. Jednotlivé sondy jsou zobrazeny na 1.1.

Velkou výhodou sektorových sond je možnost sejmut celou vyšetřovací oblast z relativně malé plochy. Toho se využívá při transtorakálním vyšetření srdce nebo při vyšetření jaterní krajiny z mezižebních prostorů.

Konvexní sonda je kombinací sektorové a lineární. Uložení elementárních měničů odpovídá lineární sondě, ale konvexní tvar plochy s měniči poskytuje obraz, který odpovídá sektorové sondě.



Obr. 1.1: Ukázka druhů ultrazvukových sond

Lineární sondy poskytují pravoúhlé zobrazení. Tento systém je tvořen velkým počtem miniaturních měničů uspořádaných do souvislé řady (linear array). Využívají se k zobrazení povrchových orgánů spolu s dopplerovskými moduly pro vyšetřování cév, kde lze použít vyšší kmitočty. [28]

1.1.3 Zobrazení

Princip dnešních ultrazvukových přístrojů spočívá na detekci a zobrazení odraženého ultrazvukového signálu (echa). K odrazu signálu dochází na rozhraní dvou prostředí. Pokud se vlna šíří homogenním prostředím, nedojde k odrazu. Pokud vlna narazí na dva různé materiály s různými fyzikálními vlastnostmi, dojde k odrazu. Tyto prostředí lze charakterizovat pomocí akustické impedance Z , která popisuje schopnost materiálu přenášet mechanické vlnění a která je dána vztahem 1.5:

$$Z = c \cdot \rho \quad (1.5)$$

kde $Z [kg \cdot m^{-2} \cdot s^{-1}]$ je akustická impedance. Množství odražené energie závisí na rozdílu akustických impedancí Z_1 a Z_2 materiálů podle vztahu 1.6:

$$R = \frac{(Z_2 - Z_1)^2}{(Z_2 + Z_1)^2} \quad (1.6)$$

kde R je koeficient odrazu (reflexe). Z této rovnice vyplývají 3 závěry:

$$\begin{aligned} R &= 0 \text{ jestliže } Z_1 = Z_2, \\ R &\rightarrow 0 \text{ jestliže } Z_1 \sim Z_2, \\ R &\rightarrow \text{ jestliže } Z_2 \gg Z_1 \text{ nebo } Z_1 \gg Z_2. \end{aligned}$$

První rovnice platí pro homogenní prostředí, ve kterém nedochází k odrazu ultrazvuku. Druhá rovnice platí pro dva materiály s podobnou akustickou impedancí. Většinou se jedná o měkké tkáně. Třetí rovnice nám ukazuje případ, kdy se jedná např. o rozhraní vzduchu a kosti. U ultrazvuku platí, že čím je větší nosná frekvence, tím větší je útlum signálu ve tkáni. Vysoké frekvence se tedy používají pro zobrazení orgánů těsně pod kůží. Pro hluboko uložené orgány se používají nižší frekvence. [23] [16] [28]

Tab. 1.1: Frekvence ultrazvuku v diagnostice [25]

Frekvence [MHz]	Aplikace
2,5	Hluboko uložené orgány, gynekologie a porodnictví
3,5	Obecná abdominální část, gynekologie a porodnictví
5,0	Vyšetření prsu, pánevní oblast, cévní vyšetření
7,5	Vyšetření prsu, štítné žlázy
10,0	Vyšetření prsu, štítné žlázy, povrchové žíly, pohybový aparát
15,0	Povrchové struktury, pohybový aparát

1.1.4 Ultrazvukové módy

A-mód(A=amplitude) je nejjednodušší zobrazení ultrazvukového signálu. Velikost zaznamenaného signálu v určitém čase lze zobrazit pomocí amplitudové výchylky na časové ose. Amplituda signálu je úměrná intenzitě odražených vln. Pomocí této metody se dají pozorovat vzdálenosti jednotlivých akustických rozhraní. Tato metoda se dnes využívá pouze v oftalmologii.

B-mód(B=brightness) využívá jasové modulace odraženého signálu, kdy daný pixel odpovídá poloze rozhraní na němž došlo k odrazu. V dnešní době se využívá dynamického B módu s rychlým snímáním a širokou stupnicí šedi (128 - 256 stupňů šedi). Podstatou tohoto zobrazení je postupné vytvoření série obrazů vyšetřované oblasti a její souvislé přehlédnutí včetně časového vývoje. Amplituda odraženého signálu odpovídá jasu pixelu. To znamená, že struktury, které dobře odráží ultrazvukový signál mají vyšší jas. Tyto struktury se nazývají echogenní. Tato metoda slouží pro zobrazení měkkých tkání.

M-mód(M=motion) se využívá při sledování pohybujících se struktur. Výsledkem je jednorozměrné zobrazení vymezené polohy určité plochy pohybující se struktury v závislosti na čase. Jedná se o jednorozměrné zobrazení, protože sledovaný

bod se pohybuje pouze po přímce dané směrem šíření ultrazvuku. Tato metoda se používá v echokardiografii.

3D-mód vytváří prostorový obraz, ale za cenu dlouhého snímání (řádově vteřiny) a počítačové rekonstrukce. Také vyžaduje speciální sondu, která je schopna generovat trojrozměrné ultrazvukové pole. Tedy 2D pole je rozvinuto kmitavým pohybem do prostoru.

Dopplerovské zobrazení je základní vyšetření, které se používá k měření rychlosti pohybujících se struktur. Měří se tedy rychlost průtoku krve. Tato metoda se dělí na systém s nemodulovanou nosnou vlnou (CW) a na systém s impulsně modulovanou nosnou vlnou (PW). [16] [28] [19] [4]

1.1.5 Kontrastní látky

Kontrastní látky se používají v mnoha lékařských aplikacích od výpočetní rentgenové tomografie až po magnetickou rezonanci. Začaly se také používat ultrazvukové kontrastní látky, díky kterým můžeme dosáhnout lepších diagnostických výsledků bez invazivního přístupu. Vyšetření, které používá kontrastní látky se nazývá dynamická kontrastní ultrasonografie (CEUS - contrast-enhanced ultrasound).

Ultrazvukové kontrastní látky zvyšují modulaci nosného signálu ve tkáni. Tedy mění primární parametrické pole a zvyšují poměr signál/šum. Používají se mikrobublinami plynu, protože rychlost šíření ultrazvuku v plynu je velmi odlišná od rychlosti šíření ve tkáni. Tento rozdíl představuje velký koeficient odrazu. Při interakci ultrazvukového signálu s bublinami dochází k rezonanci a zvyšuje se tedy intenzita odražených vln. Použití mikrobublinek je podmíněno jejich průchodem plicními kapilárami a dostatečnou stabilitou v krevním řečišti. Velikost mikrobublin se pohybuje v rozmezí 1-10 mikrometrů, jsou tedy srovnatelné s velikostí červených krvinek. [7] Echokontrastní látky se v krevním řečišti chovají podle energie ultrazvukového stimulu, tedy podle hodnoty mechanického indexu. Při nízké energii ($MI = 0.2$) se ultrazvukové impulsy odrážejí. Při střední energii ($MI = 0.2-1.0$) se bubliny dostávají do rezonance s frekvencí ultrazvukové vlny. Při vyšší energii ($MI = 1.0$) bubliny zanikají (dojde ke kavitaci a přeměně na teplo). [28]

Technika s vysokým MI (> 1) je také nazývána jako stimulovaná akustická emise (SAE). Pokud dojde ke zvýšení MI v pozdní fázi, dojde k destrukci bublin, která se následně projeví jak v šedém B zobrazení tak i v barevném dopplerovském obraze. Jako kontrastní látky se používají rychlorozpustné plyny.

Technika s nízkým MI (< 1) je založena na principu rezonanci bublin s možností časového záznamu tří vaskulárních fází. Při použití této metody dojde k vyplavení (wash-in) kontrastní látky do jednotlivých tkání a následného vyplavení (wash-out) kontrastní látky s cílem zpřesnit diagnostiku. Jsou používány pomalu rozpustné plyny. [28]

Kontrastní látky v ultrasonografii můžeme rozdělit do tří skupin podle časové stability. První skupinou jsou látky přetrvávající pouze několik sekund. Do této skupiny patří např. protřepaný fyziologický roztok. V druhé generaci se zkoumala velikost a stabilizace mikrobublin, protože tyto vlastnosti určují jejich účinnost. Tyto látky vydrží minuty až několik desítek minut. Získávají se přidáním cukerného nosiče a stabilizátoru do fyziologického roztoku. Tyto látky dopomohly k rozšíření kontrastních látek v ultrasonografii. Je možné je detekovat v zobrazení B-módu až po levou stranu srdce, ale jakmile dojde k jeho rozředění v systémovém oběhu, je potřeba více citlivá technika. Třetí generace kontrastních látek splývá s druhou generací díky technickému vývoji (schopnost detekovat vyšší harmonické frekvence). Díky tomuto vývoji bylo možné zachytit odrazivost struktur i při dvourozměrném zobrazení. [11] [12]

1.1.6 Perfuzní zobrazení

Zobrazení prokrvení (perfuze) je významným diagnostickým nástrojem pro detekci ischemických a onkologických onemocnění. Aplikace kontrastních látek v ultrasonografii se provádí dvěma způsoby. Jedním z nich je jednorázová injekce bolusu (jednorázová dávka dané látky). Druhým způsobem je konstantní infuze, při které se používá plánované ničení mikrobublin s následným znovu zaplněním oblasti zájmu (vytváření negativního bolusu) a následného sledování průtoku krve. Občas se předem používá injekce bolusu, protože ustálení indikátoru na stálé hladině trvá určitý čas. Tato metoda se používá pro lepší zobrazení cévního kmene, hemodynamicky významné stenózy, uzávěry nebo aneurysmata.

Po aplikaci kontrastní látky se snímá intenzita odraženého signálu v daném objemu tkáně (ROI) v čase t . Intenzita odraženého signálu je přímo úměrná koncentraci kontrastní látky. Metody perfuzního zobrazení se dají popsat několika funkcemi (parametry).

- AUC (Area under the time-intensity curve) - plocha pod křivkou
- MTT (the mean transit time) - střední doba průchodu
- t_p (the time to peak intensity) - čas maximální intenzity

Tyto parametry jsou závislé na průtoku krve a objemu krve v oblasti zájmu. Pokud je tedy známo množství kontrastní látky a parametry AUC a MTT, je možné vypočítat

průtok krve a následně i objem krve v oblasti zájmu podle 1.7 a 1.8:

$$F = m \cdot (AUC)^{-1} \quad (1.7)$$

$$V = F \cdot MTT \quad (1.8)$$

kde F je průtok krve, m je množství kontrastní látky (bolusu) a V je objem krve.[7]
[13]

2 MATEMATICKÉ MODELÝ

2.1 Matematické modely

Matematickým modelem se rozumí popis vlastností, vztahů a struktur zkoumaného objektu pomocí matematických funkcí, rovnic a konstant. Tyto kroky vedou k realizaci počítačového modelu, který je realizován pomocí výpočetní techniky. Tento model následně umožňuje simulovat chování systému. Tyto modely se dnes využívají např. v biologii.

Modelování je označení pro soubor aktivit vedoucích k vývoji matematického modelu, který představuje reálný systém. Tento model představuje zjednodušený abstraktní popis reálného systému. **Simulace** označuje soubor aktivit sloužících k ověření korektnosti modelu a získání nových informací o činnosti a funkci reálného systému. Modely nikdy přesně neodpovídají reálnému systému, ale mohou být velmi užitečné. [18][27]

2.1.1 Klasifikace matematických modelů

Principiálně rozeznáváme tři základní přístupy k modelování.

- model je založen na teoretických znalostech o řešeném problému nebo na zkoumání experimentálních dat
- struktura modelu odpovídá struktuře reálného systému
- podle složitosti modelu

Apriorní a aposteriorní návrh modelu

Apriorní (teoretické) modely jsou formulovány na základě již existujících znalostí o funkci modelovaného systému. Např. model metabolismu glukózy může být založen na experimentech in-vitro a na následných teoretických předpokladech o reakci na živém organismu.

Oproti tomu *aposteriorní* modely vznikají na základě experimentu. Model musí být sestaven tak, aby co nejlépe aproximoval získaná data. Tedy u modelu chování glukózy měříme její koncentraci v čase po jejím vstříknutí do organismu. Tyto modely se označují jako černá skříňka nebo jako šedá skříňka. Označení odpovídá znalostem vnitřní struktury zkoumaného reálného systému.

Popisný a mechanistický návrh modelu

Popisné modely obsahují matematické funkce a rovnice, které popisují experimentální data. Parametry jednotlivých rovnic neodpovídají prvkům, parametrům a proměnným reálného systému. Tyto modely jsou založeny na předpokladech a musí být dále ověřeny. Naměřená data nejsou plně využita a jejich platnost je omezená na známá data - popis dat není jejich vysvětlení. [27]

Mechanistické modely jsou založeny na znalosti procesů a mechanismů, které probíhají v zkoumaném reálném systému. Funkce modelu se vztahují k veličinám reálného objektu a mohou vysvětlit jeho strukturu a funkci. Nevýhodou je, že musí být známá struktura systému. Výhodou je, že můžeme navrhnout experiment podle simulačních dat.

Deterministické a stochastické modely

Deterministické modely mají jasně definované parametry a průběhy signálů. Tyto modely mají při stejném nastavení vždy stejný výsledek.

Stochastické modely na rozdíl od deterministických modelů mají jednotlivé proměnné zadány jako náhodné či jsou zadány s určitou pravděpodobností výskytu jevu. Používají se pro stanovení dynamických rychlostních parametrů tvorby. Většina biologických systémů zahrnuje oba přístupy.

Statické a dynamické systémy

Statickými modely nazýváme systémy, jejichž chování je nezávislé na vstupech v minulosti. Jsou to systémy bez paměti.

Naopak **dynamické** systémy jsou systémy s pamětí. Jejich výstup vždy závisí na jejich chování v minulosti. Např. u fyzikálních systémů mluvíme o dynamice systému.

Diskrétní a spojité systémy

Diskrétní systémy jsou popsány vstupy, výstupy či stavy jen v určitých časových intervalech. Ve většině případů se jedná o ekvidistantní vzorkování.

Spojité modely jsou popsány souvisle v čase. Signály jsou tedy spojité funkce času.

Lineární a nelineární modely

Lineární a **nelineární** modely lze rozlišit podle toho, zda u nich platí princip superpozice. Pro biologické systémy a všechny reálné systémy obecně platí, že jsou nelineární. Linearita je pouze aproximací s určitou (dostačující) přesností.

2.1.2 Farmakokinetické modely

Základem této kapitoly je farmakokinetika, která se zabývá distribucí léčiva v organismu v závislosti na čase. Léčivo je v organismu postupně distribuováno, vstřebáváno, metabolicky transformováno a vyloučeno z organismu. Pomocí farmakokinetických modelů můžeme zjistit časový průběh léčiva v organismu. Tyto modely se mohou užít pro zkoumání distribuce nejen léčiva, ale i např. živin, radionuklidů, kontrastních látek nebo toxinů.

Distribuce léčiva se dělí do tří fází:

1. Resorpce - přesun léčiva do tkání z různých míst, podle způsobu podání
2. Distribuce - rozvedení léčiva do tkání
3. Eliminace - vyloučení léčiva z organismu dechem, ledvinami či biotransformací

Způsoby podání léčiva se dělí na:

1. Přímo do krve
 - rychlá nitrožilní aplikace
 - dlouhodobá nitrožilní aplikace
2. Ústy (perorálně)
3. Do svalu (intramuskulárně)
4. Pod kůží (subkutánně)

2.2 Základní farmakokinetické parametry

Kinetika léčiv se dá popsat různými proměnnými. Dělí se na **primární** a **sekundární** parametry. Mezi primární parametry patří:

Distribuční objem V_d

Parametr udávající poměr mezi množstvím léčiva v těle a jeho koncentrací v krevní plasmě. Je dán vztahem 2.1:

$$V_d = \frac{M_t}{c_p} \quad (2.1)$$

kde M_t [mg] je množství léčiva v těle a c_p [mg · l⁻¹] je koncentrace v krevní plasmě. Nejedná se o reálný objem, pouze udává objem, ve kterém by se muselo rozptýlit léčivo, aby jeho koncentrace byla stejná jako v krevní plasmě. Tělo zde představuje jeden kompartment s homogenním rozložením léčiva. Distribuční objem se většinou

udává relativně vzhledem k hmotnosti pacienta, tedy v $l \cdot kg^{-1}$. Tento parametr se také používá k výpočtu nárazové (nasycovací) dávky D_N podle vztahu 2.2:

$$D_N = V_d \cdot c_p \quad (2.2)$$

Clearance CL

Veličina clearance určuje rychlost eliminace vzhledem ke koncentraci látky v krvi. Je dána vztahem 2.3:

$$CL = \frac{v_e}{c_p} \quad (2.3)$$

kde v_e je rychlost eliminace léčiva a c_p je koncentrace látky v krvi. V procesu eliminace jsou zahrnuty všechny eliminační procesy, jako je eliminace ledvinami, střevy, játry a chemickými přeměnami. Proměnná rychlost eliminace je dána jako množství léčiva za čas. Pro rychlost eliminace pak platí vztah 2.4:

$$v_e = k \cdot M_t(t) = k \cdot c_p(t) \cdot V_d \quad (2.4)$$

kde k je rychlostní konstanta eliminace. Z této rovnice lze vyjádřit clearance pomocí rychlostní konstanty a distribučního objemu jako 2.5:

$$CL = k \cdot V_d \quad (2.5)$$

Mezi sekundární parametry patří:

Biologický poločas eliminace ($t_{\frac{1}{2}}$)

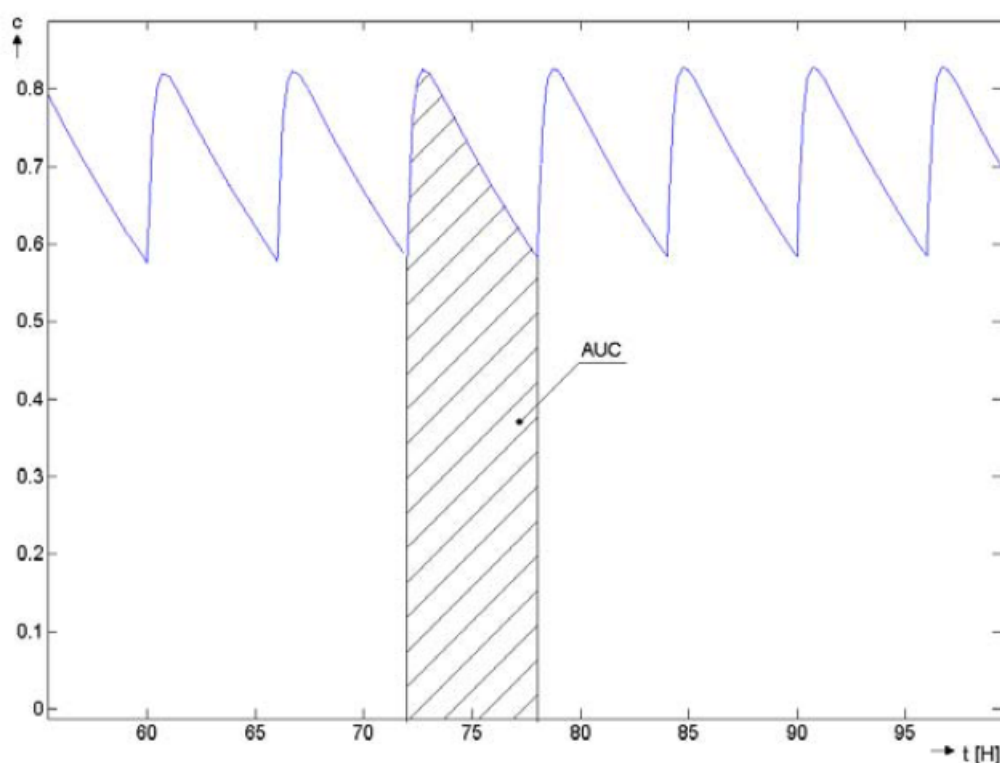
Tento parametr udává čas, za který dojde k poklesu koncentrace léčiva v organismu na polovinu počáteční hodnoty. Pro lineární jednodukompartmentový model klesá koncentrace léčiva exponenciálně. Proto lze odvodit :

$$t_{\frac{1}{2}} = \frac{\ln 2}{k} = \frac{0,693}{k} = \frac{0,693 \cdot V_d}{CL} \quad (2.6)$$

kde k je rychlostní konstanta eliminace. Z biologického poločasu rozpadu lze odvodit kumulaci léčiva při opakovaných dávkách, nebo čas potřebný k ustálenému stavu koncentrace léčiva v organismu při podání léčiva pomocí infuze.

Plocha pod křivkou(AUC)

Jedná se o plochu pod křivkou, která je vymezena průběhem koncentrace látky v krevním oběhu v závislosti na čase. Při jednorázovém podání se jedná o celý průběh křivky. Při infuzním podání léčiva se jedná o obsah pod křivkou při jednom dávkovacím cyklu.



Obr. 2.1: Plocha pod křivkou AUC, převzato z [18]

Biologická dostupnost F

Tento parametr udává množství podaného léčiva, které se dostane v aktivní podobě do krevního oběhu. Vyjadřuje relativní část dávky a nabývá hodnot 0 až 1. Při intravenózním podání je biologická dostupnost rovna 1, protože se léčivo aplikuje přímo do krevního oběhu. Při extra-vasculárním podání se do krevního oběhu dostane jen část léčiva. Základem je vstřebatelnost látky, degradace látky a tzv. efekt prvního průchodu játry, kdy dojde k největšímu snížení množství látky v krvi.

Určujeme tedy absolutní biologickou dostupnost jako poměr plochy pod křivkou po podání uvažované látkové formy AUC, vztažené k ploše pod křivkou při intravenózním podání stejné látky $AUC_{i.v.}$ 2.7:

$$F_{abs} = \frac{AUC}{AUC_{i.v.}} (\cdot 100\%) \quad (2.7)$$

Relativní biologická dostupnost se určuje pro jiný druh podání lékové formy než intravenózní.

2.2.1 Jednokompartmentové farmakokinetické modely

Jedná se o nejjednodušší model, kdy je použit pouze jeden kompartment. Lidské tělo je tedy popsáno jako jeden homogenní objem. Použije se v případě, kdy dochází k rychlé změně koncentrace látky v orgánech při změně koncentrace v krvi.

Pro nitrožilní injekci je předpoklad okamžitého rozptýlení léčiva. Množství léčiva v kompartmentu lze popsat jako 2.8:

$$\frac{dM(t)}{dt} = -k_e \cdot M(t) \quad (2.8)$$

$$M(0) = M_0 \quad (2.9)$$

kde M_0 je podaná dávka a je podmínkou rovnice, K_0 je rychlostní konstanta eliminace léčiva.

Rovnici pro koncentraci léčiva v krevní plasmě $C(t)$ lze popsat jako 2.10:

$$\frac{dc(t)}{dt} = -k_e(t) \quad (2.10)$$

$$c(t) = c_0 \quad (2.11)$$

kde c_0 je počáteční koncentrace léčiva v krevní plasmě, vypočtená jako dávka M_0 dělená distribučním objemem V_d .

Z řešení rovnic 2.8 a 2.10 lze časový průběh množství látky a koncentrace odvodit jako :

$$M(t) = M_0 \cdot e^{-k_e t} \quad (2.12)$$

$$c(t) = c_0 \cdot e^{-k_e t} \quad (2.13)$$



Obr. 2.2: Jednokompartmentový model distribuce léčiva

2.2.2 Vícekompartmentové farmakokinetické modely

Tyto modely se používají pro podrobnější modelování distribuce léčiva. Organismus je zde popsán více kompartmenty. Např. pro lidské tělo jsou dva kompartmenty, jeden

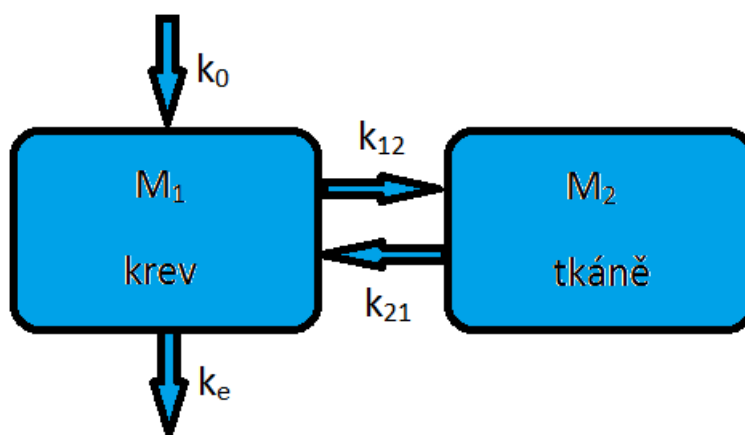
pro trávicí trakt ze kterého je vstřebáváno léčivo do krevního oběhu (tj. druhého kompartmentu). Pro tento model platí rovnice 2.14,2.15:

$$\frac{dM_1}{dt} = k_0 - k_{12} \cdot M_1 \quad (2.14)$$

$$\frac{dM_2}{dt} = k_{12} \cdot M_1 - k_e \cdot M_2 \quad (2.15)$$

Kde konstanta k_0 je dávka léčiva, rychlostní konstanta k_{12} udává rychlost vstřebávání látky z trávicího traktu do krevního oběhu. Koncentrace léčiva v krvi stoupá do doby, kdy dojde k vyrovnaní rychlosti absorpce a vylučování.

Další dvoukompartmentový model reprezentuje kompartmenty krevního oběhu a tkání. Léčivo přechází zpětně do tkáně a eliminace probíhá pouze přes krevní oběh.



Obr. 2.3: Vícekompartmentový model s rozdělením na krevní oběh a tkáň

2.3 Matematické modely pro perfuzní zobrazení

2.3.1 Log-normální rozložení

Od poloviny dvacátého století se začalo log-normální rozložení používat jako model pro výpočet dilučních křivek. Pomocí logaritmicko-normálního rozdělení lze popsat např. rozdělení tělesné hmotnosti v populaci, délku inkubační doby infekčního onemocnění nebo hladinu glukózy v krevním séru. V neposlední řadě jím lze charakterizovat i krevní parametry jako koncentraci echo-kontrastní látky v daném objemu. Náhodná veličina I má logaritmicko-normální rozdělení právě tehdy, když veličina $Y = \ln(I)$ má normální rozdělení.

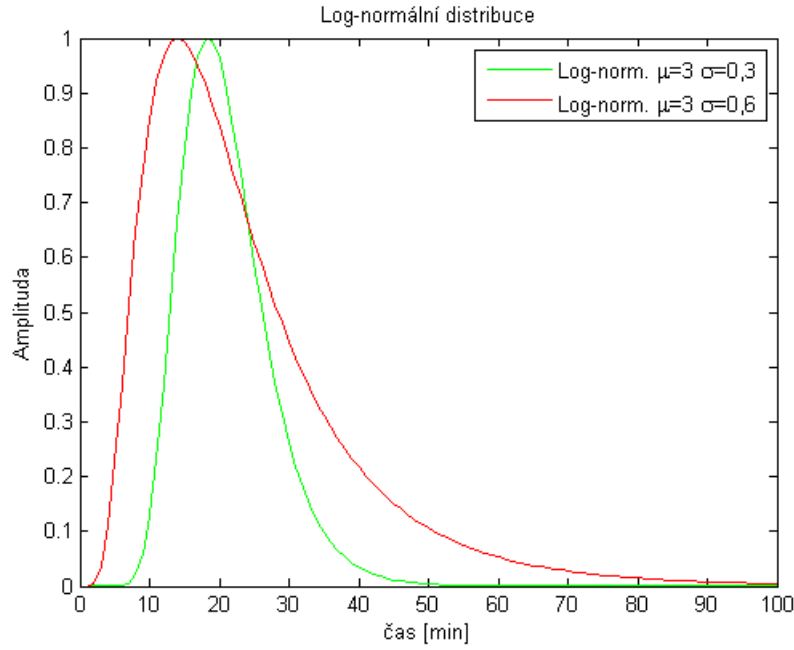
Log-normální distribuční funkce se zpožděním $t(0)$ je dána jako 2.16:

$$I(t) = \frac{AUC}{\sqrt{2\pi}\sigma \cdot (t - t_0)} \cdot e^{-\frac{(\ln(t-t_0)-\mu)^2}{2\cdot\sigma^2}} + I_0; t > t_0 \quad (2.16)$$

kde $I(t)$ je intenzita odraženého signálu (úměrného koncentraci kontrastní látky) v závislosti na čase. Proměnné μ a σ jsou průměr a směrodatná odchylka. Hodnota μ odpovídá horizontální nastavení křivky a hodnota σ odpovídá za nastavení strmosti výsledné křivky. Hodnota I_0 odpovídá izolínii (DC offset). Očekávaný čas zaznamenání bolusu je nastaven na nulu osekáním dat a resetováním hodnoty t . Parametr t_0 je malou úpravou k původnímu času rozložení. Tento parametr je získán z kritériální funkce. Je důležitý, protože je těžké rozeznat první příchod kontrastní látky kvůli malému poměru signál-šum. Na grafu 2.4 je vidět log-normální distribuce pro různé nastavení konstant σ a μ . [13]

Dále je možné definovat MTT a t_p jako 2.17:

$$MTT = e^{\frac{\mu+\sigma^2}{2}}; t_p = e^{\mu-\sigma^2} \quad (2.17)$$



Obr. 2.4: Průběh log-normální distribuce

2.3.2 Erlang rozložení

Tato funkce je odvozena za předpokladu, že konstantní průtok krve může být modelován z n homogenních kompartmentů o stejném objemu V_c , přičemž každý kompartment má pouze jeden vstup a jeden výstup krve. Po injekci indikátoru je rychlost změny koncentrace $C_i(t)$ v kompartmentu i ($i = 1, \dots, n$) dána následujícími diferenciálními rovnicemi 2.18 a 2.19:

$$\frac{dC_1}{dt} = \frac{m}{V_0}\delta(t) - \frac{F}{V_0}C_1 \quad (2.18)$$

$$\frac{dC_i}{dt} = \frac{F}{V_0}[C_{i-1}(t) - C_i(t)]; \quad i = 2, \dots, n, \quad (2.19)$$

kde $\delta(t)$ je Diracova funkce. Platnost těchto rovnic je založena na předpokladu, že dojde k úplnému smíchání indikátoru a krve v každém kompartmentu. Další podmínkou je jednosměrný přechod mezi kompartmenty. Po vyřešení rovnic 2.18 a 2.19 dostaneme $C_n(t)$, které je lineárně závislé na $I(t)$:

$$I(t) = \frac{AUC}{\beta^n(n-1)!}(t-t_0)^{n-1}e^{\frac{-(t-t_0)}{\beta}} + I_0 \quad (2.20)$$

kde $\beta = V_c F^{-1}$ se nazývá parametr rychlosti, protože je roven časové konstantě pro odezvu jednoho kompartmentu. Počet kompartmentů n je znám jako parametr tvaru, protože šikmost rozdělení je rovna $2/n$, z čehož vyplývá, že $I(t)$ se stává symetričtějším se zvyšujícím se počtem kompartmentů.

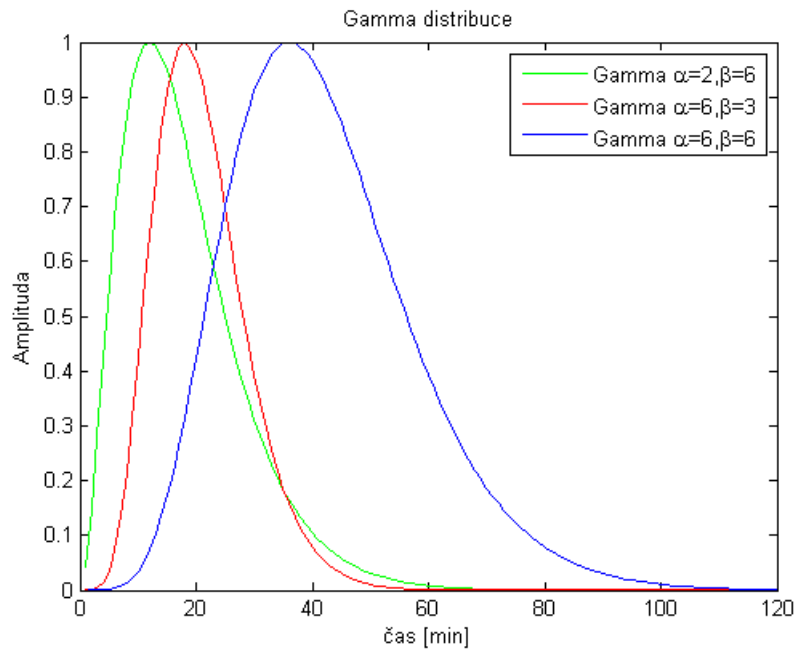
2.3.3 Gamma rozložení

Jednoduchost gamma funkce je dobře použitelná pro výpočet intenzity jasových signálů pro CEUS. Gamma funkce je nastavitelná pomocí parametrů α a β . Při konstantním parametru β a zvyšujícím se parametru α se stává výsledná křivka více symetrická (méně zkosená). Pro konstantní parametr α a zvyšující se parametr β dojde ke zvýšení parametru wash in a wash out. Gamma funkce je dána vztahem 2.21:

$$I(t) = \frac{AUC}{\beta^{\alpha+1}\Gamma(\alpha+1)}(t-t_0)^\alpha e^{-\frac{(t-t_0)}{\beta}} + I_0 \quad (2.21)$$

kde $\alpha - 1$, člen $\beta^{\alpha+1}\Gamma(\alpha+1)$ normalizuje gamma funkci tak, že $(I(t) - I_0)$ je pravděpodobnostní distribuce, která integruje k jedné, když AUC se rovná 1. Výhodou může být z praktických vlastností gamma funkce pro výpočet MTT a t_p . 2.22:

$$MTT = \beta(\alpha + 1), \quad t_p = \cdot\beta \quad (2.22)$$



Obr. 2.5: Průběh gamma distribuce

Typický průběh gamma funkce pro různá nastavení konstant α a β .

2.3.4 LDRW

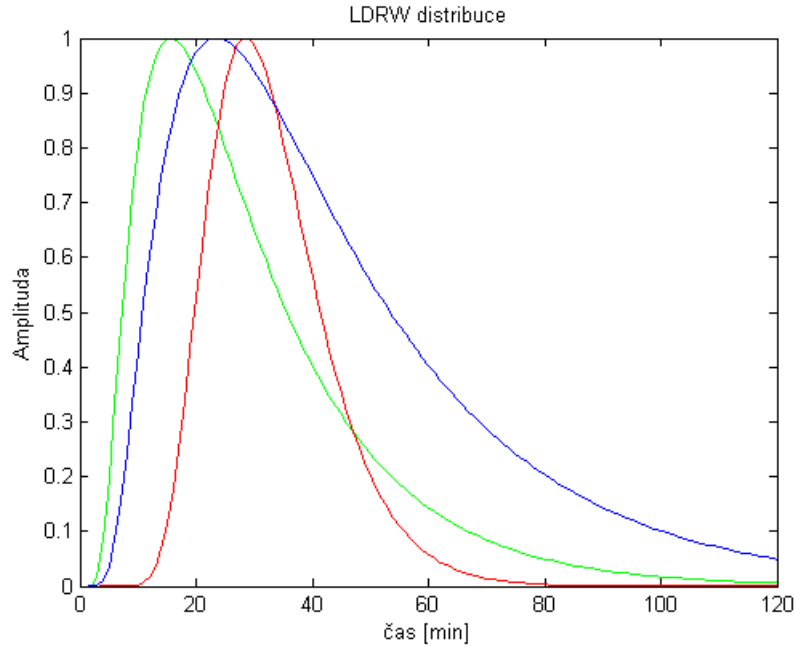
Neboli Local Density Random Walk je funkce, která využívá pohybu kontrastních látek v určité (konečné) délce cévy. Pohyb kontrastní látky přes cirkulační systém je dán kombinací podélné difuze a lineárního proudění. Pro jednorázovou dávku bolusu je možné získat koncentraci $C(x, t)$, která je lineárně závislá na intenzitě odraženého signálu z CEUS. Poté dostaneme $I(t)$, které je dáno jako 2.23:

$$I(t) = AUC \cdot \left(\frac{e^\lambda}{\mu}\right) \sqrt{\frac{\mu\lambda}{(t-t_0)2\pi}} \cdot \exp\left[-\frac{1}{2}\lambda\left(\frac{\mu}{(t-t_0)} + \frac{(t-t_0)}{\mu}\right)\right] + I_0 \quad (2.23)$$

kde $\mu \equiv x_0/v$ (x_0 je vzdálenost mezi vstupem a výstupem sledované oblasti ROI) je střední doba potřebná pro průchod mikrobublin zájmovou částí x_0 . Parametr λ je definován jako $\lambda \equiv \mu v^2/2D$ a strmost křivky je rovna λ^{-1} .

Pro LDRW model je možné definovat MRT (mean residence time), což je čas, po který mikrobubliny zůstanou ve sledované oblasti. Jako u ostatních modelů můžeme definovat MTT a t_p jako:

$$MRT = \mu(1 + \frac{1}{\lambda}), \quad MTT = \mu, \quad t_p = (\frac{\mu}{2\lambda})(\sqrt{1 + 4\lambda^2} - 1) \quad (2.24)$$



Obr. 2.6: Průběh LDRW distribuce

3 OPTIMALIZACE PRO MODELOVÁ DATA

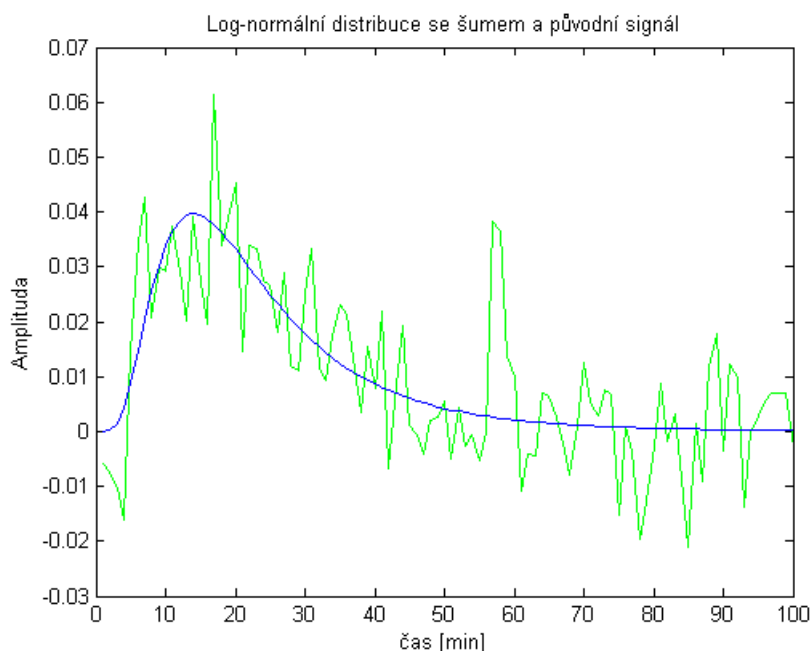
Pro vytvoření modelů a grafů bylo použito programové prostředí MATLAB R2013A. Jednotlivé modely byly vytvořeny v tomto prostředí a generovány podle výše zmíněných vzorců.

3.1 Modelování šumu

Aby bylo možné zhodnotit proložení křivek, bylo nutné původní křivky zašumět. K tomu byla použita funkce *randn*, která má 2 vstupy. Jde o normalizované normální rozložení s nulovou střední hodnotou a směrodatnou odchylkou rovnou 1.

$$S = randn(y, x) \times k \quad (3.1)$$

Jedním ze vstupů je počet řádků y a druhým je počet sloupců x . Počet sloupců musí být roven délce signálu, aby byli zašuměny všechny vzorky. Problém je, že tato funkce generuje čísla v intervalu, který by po přičtení k původnímu signálu úplně tento signál znehodnotil. Proto jsou tato generovaná čísla vynásobena konstantou $k = 0,01$, aby odpovídala rozsahu signálu. Následně jsou oba vektory hodnot sečteny a dostaneme signál se šumem. Ukázka takového signálu je zobrazena na grafu 3.1 :



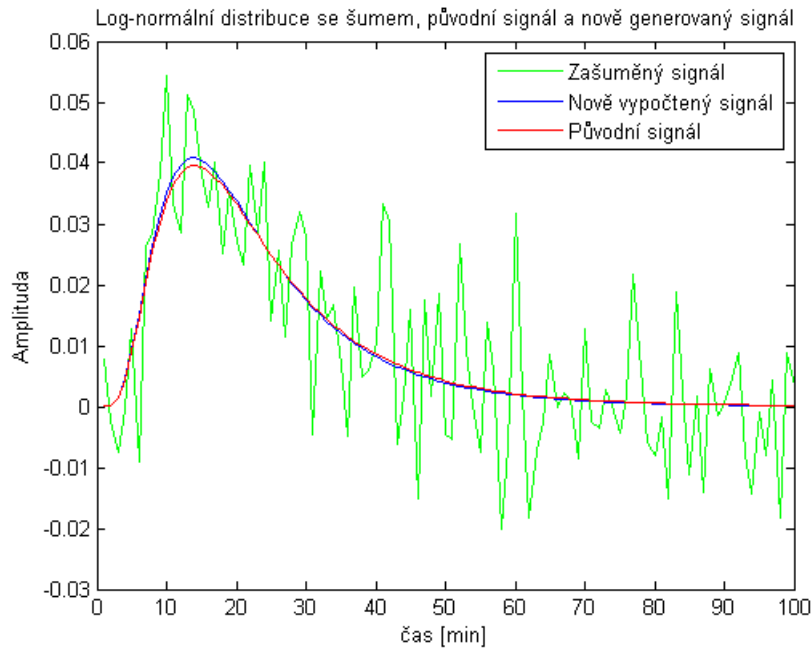
Obr. 3.1: Průběh log-normální distribuce se šumem a původní signál

3.2 Popis funkcí

Pro proložení dat modelem byla použita funkce `lsqcurvefit`. Jedná se o metodu nejmenších čtverců, která hledá minimum rozdílu původní funkce a generované funkce. Funkce se zadává ve tvaru `v=lsqcurvefit(@fnc,x_0,xdata,ydata)`, kde hodnoty výstupu v jsou koeficienty námi zadané rovnice. Pro proložení křivky je volána funkce, která obsahuje tvar námi požadované rovnice pro proložení, k tomu slouží člen `@fnc`. Člen x_0 jsou odhady koeficientů pro hodnoty $xdata$, aby co nejlépe odpovídala hodnotám $ydata$. Tato funkce provádí matematický výpočet minimalizace součtů kvadrátů odchylek podle vztahu 3.2:

$$\min_x \| F(x, xdata) - ydata \|_2^2 = \min_x \sum_i (F(x, xdata_i) - ydata_i)^2 \quad (3.2)$$

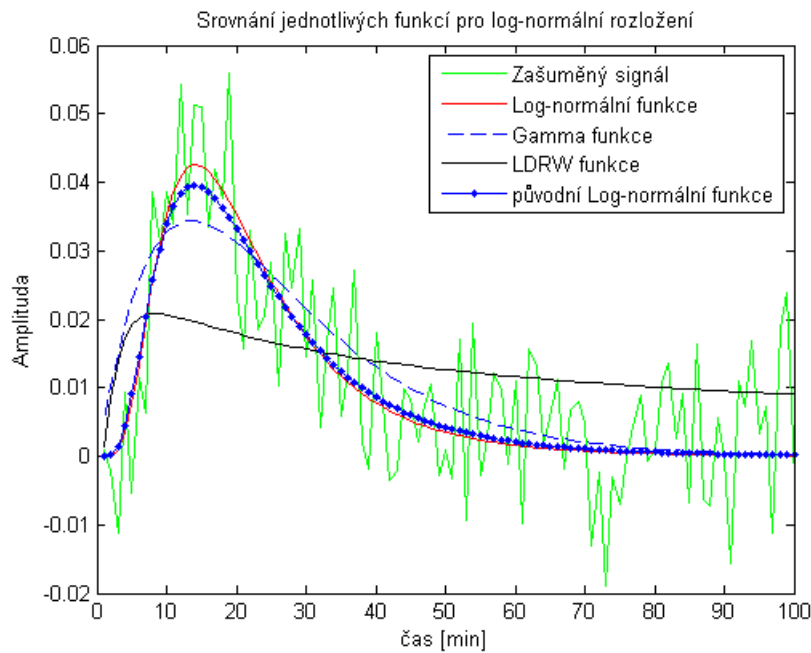
Po vypočtení a dosazení koeficientů dostaneme graf 3.2:



Obr. 3.2: Průběh log-normální distribuce se šumem a nově generovaný signál

Zde je patrné, že nově generovaný signál s vypočtenými parametry se téměř blíží k původnímu signálu.

Aby bylo možné kvantitativně měřit kvalitu proložení, je nutné jednotlivé průběhy sloučit. To je možné provést tak, že je vygenerován signál, ale poté je proložen jinou funkcí než tou, jíž byl generován. Na 3.3 je vidět původní signál, generovaný jako log-normální rozložení. Dále jsou zde vykresleny i ostatní funkce. Je patrné, že nejlepší výsledek je dosažen log-normální proložením, následně gamma funkcí a nejhorší výstup poskytuje LDRW funkce [10].



Obr. 3.3: Srovnání jednotlivých funkcí pro log-normální vstup

3.3 Výstupy

Pro zhodnocení výsledných generovaných signálů je počítán koeficient, který určuje odchylku nově generovaných průběhů od původního průběhu. Tímto lze určit kvalitu jednotlivých funkcí pro proložení jiných vstupních dat. Tento koeficient je vypočítán podle vztahu 3.3 :

$$S = \left(\frac{1}{n} \sum |f(n) - g(n)| \right)^2 \quad (3.3)$$

kde n je počet vzorků signálu, $f(n)$ a $g(n)$ jsou původní a nově generovaný signál. Hodnoty koeficientu pro jednotlivé vstupní signály a jednotlivé generované křivky jsou v tabulce 3.3

Tab. 3.1: Srovnání koeficientu pro jednotlivé funkce

	Hodnoty koeficientu pro absolutní hodnotu rozdílu		
Vstupní signál	Log-normální	Gamma	LDRW
Log-normální	0,0011	0,0014	0,0070
Gamma	0,0150	0,0003	0,0115
LDRW	0,0264	0,0045	0,0002

Pro kvantitativní zhodnocení výsledků proložení jednotlivými funkcemi byl vypočten Pearsonův korelační koeficient mezi jednotlivými funkcemi. Byl použit vzorec

3.4 :

$$\gamma_{xy} = \frac{\sum_{n=0}^{N-1} x(n)y(n)}{\sqrt{\sum_{n=0}^{N-1} x^2(n) \sum_{n=0}^{N-1} y^2(n)}} \quad (3.4)$$

Pro korelační koeficient γ_{xy} platí, že hodnota 0 znamená to, že jsou signály nekorelované a tudíž mají menší podobnost. Hodnoty -1 a 1 určují, že signály jsou lineárně závislé a tudíž jsou si podobné.

Tab. 3.2: Srovnání Pearsonova korelačního koeficientu pro jednotlivé funkce

	Hodnoty Pearsonova korelačního koeficientu		
Vstupní signál	Log-normální	Gamma	LDRW
Log-normální	0,0274	0,0229	0,0132
Gamma	0,0400	0,0312	0,0220
LDRW	0,0182	0,0137	0,0128

Pokud jsou hodnoty blíže k 0, nemusí to znamenat že nejsou korelované, jen u nich neplatí lineární závislost. Z tabulky 3.3 tedy vyplývá, že jednotlivé funkce nejsou lineárně závislé, až na log-normální funkci.

4 GLOBÁLNÍ OPTIMALIZACE

Metaheuristické algoritmy, jako jsou optimalizace částicemi roje nebo simulované žíhání se v dnešní době dostávají do popředí pro řešení optimalizačních problémů. Drtivá většina těchto metod jsou založeny na chování biologických nebo fyzikálních systémů. Například optimalizace rojem částic byla vyvinuta na základě chování hejna ptáků a ryb.

Optimalizační algoritmy slouží k nalezení minima či maxima účelové funkce tak, že je hledána optimální kombinace jejích argumentů. Existuje mnoho různých rozdělení. Jedno z nich je:

- Enumerativní
 - Deterministické
 - * Greedy
 - * Hill-Climbing
 - * Branch a Bound
 - * Depth-First
 - * Breadth-First
 - * Best-First
 - * Calculus Based
- Globální prohledávání a optimalizace
 - Stochastické
 - * Random Search-Walk
 - * Simulated Annealing
 - * Monte Carlo
 - * Tabu Search
 - * Evolutionary Computation
 - * Stochastic Hill-Climbing
 - Smíšené
 - * Mathematical programming
 - * Ant Colony Optimization
 - * Immune System Methods
 - * Memetic Algorithms
 - * Scatter Search a Path Relinking
 - * Particle Swarm
 - * Genetic Algorithms
 - * Differential Algorithms
 - * SOMA

Použití jednotlivých optimalizačních metod je závislé na druhu úlohy a prohledávané oblasti účelové funkce. [26]

4.1 Simulované žíhání

Jednou z metod globální optimalizace je metoda zvaná simulované žíhání (*simulated annealing*). Metoda je variantou horolezeckého algoritmu, v němž jsou heuristické kroky, které směřují k horšímu řešení, řízeny s určitou pravděpodobností. Metoda má základ v simulování fyzikálních procesů probíhajících při pomalém chlazení kovu. Při zahřátí kovu jsou atomy vybuzeny do vyšších energetických stavů. Při následném pomalém ochlazování postupně zůstávají ve svých nejnižších energetických stavech. Tento proces se používá při odstraňování vad materiálů. Tento algoritmus poprvé matematicky popsal N. Metropolis v roce 1953. Algoritmus simuluje malou změnu pozice atomu, která má za výsledek změnu energie. Pokud je změna negativní (dojde ke snížení energie) je nová pozice přijata. Pokud je změna pozitivní (došlo ke zvýšení energie) pak může být nová pozice přijata s pravděpodobností danou Boltzmanovým faktorem 4.1:

$$P = e^{\left(\frac{-\Delta E}{k_0 T}\right)} \quad (4.1)$$

kde ΔE je změna funkční hodnoty mezi novou a předešlou hodnotou, T je parametr odpovídající teplotě tělesa. [22]

Algoritmus se řídí těmito pravidly:

- Jestliže $f(x') \leq f(x)$, pak je nové řešení přijato s pravděpodobností rovnou jedné. Nový vektor je tedy akceptován v další iteraci.
- Jestliže $f(x') \geq f(x)$, pak je pravděpodobnost přijetí nového řešení rovna Boltzmanové proměnné. Nový vektor se prosadí i v případě horšího řešení.
- Jestliže, je na začátku algoritmu nastavena velká hodnota T , pak je i vyšší šance na to, že bude přijato i horší řešení.
- V průběhu algoritmu se hodnota T snižuje a tím se také snižuje pravděpodobnost přijetí horšího řešení.

[21, 24]

4.2 Genetické algoritmy

Samotný základ genetických algoritmů (GA) vychází z teorie evolučních procesů probíhajících v biologických systémech. Tuto teorii popsal Charles Darwin a je podle něj také pojmenována jako Darwinova teorie. Z té vyplývá, že v přirozeném výběru přežívají pouze jedinci s největší schopností adaptace. Míra adaptace je tzv. 'fitness' jedince. V biologii je fitness funkce chápána jako schopnost jedince přežít a reprodukovat se. Pokud se na to podíváme z ohledu biologické evoluce, pak se jedná o změnu genetické informace v průběhu mnoha generací k vyšším hodnotám fitness funkce. Z toho vyplývá, že jedinec s vyšší hodnotou fitness funkce má vyšší pravděpodobnost

reprodukce. Ke změně genetické informace se dále uplatňuje mutace a křížení. Většina genetických algoritmů využívá zápis čísel v binární podobě. Tento systém má tu nevýhodu, že je omezený na určitý, předem definovaný kvantizační krok. V této práci se zaměřím na spojité genetické algoritmy, které využívají reprezentaci hodnot s pohyblivou řádovou čárkou. Pro vyjádření čísla se používají dvě složky: mantisa M a exponent E , kdy číslo má výslednou podobu $M \cdot 10^E$.

4.2.1 Generování počáteční populace

Počáteční populace se volí náhodně podle velikosti prohledávaného prostoru. Obvykle se jedná o stovky či tisíce jedinců (chromozómů). Pro N -rozměrný optimalizační problém je dáno N_{var} proměnných. Tedy

$chromozóm = \{p_1, p_2, p_3, \dots, p_{N_{var}}\}$ a následně je pro každý chromozóm vyčíslena fitness funkce f jako: $cost = f(chromozóm) = f(p_1, p_2, p_3, \dots, p_{N_{var}})$. Nová populace je generována maticí o velikosti $N_{pop} \cdot N_{var}$, kde N_{pop} je počet jedinců v generaci.

4.2.2 Genetické operátory

Při práci s GA se využívají jako genetické operátory selekce, reprodukce a mutace. **Selekce** se provádí různými způsoby. Většina těchto způsobů je založena na principu pravděpodobnosti výběru jedince s vyšší fitness hodnotou. Proto se jedinci s nejvyššími hodnotami reprodukuje více a jsou více obsaženi v dalších generacích, zatímco jedinci s nízkou fitness hodnotou se reprodukuje jen málo. Ale i tito jedinci mají určitou možnost postupu do další generace, aby se zamezilo uvíznutí v lokálním extrému. Zde si uvedeme některé z nich:

Ruletová selekce

Pravděpodobnost výběru jedince do další generace je přímo úměrná jeho fitness hodnotě. Průběh této operace si můžeme představit z jejího názvu. Vezměme ruletu a jednotliví jedinci jsou číselné hodnoty, na kterých se může kulička v ruletě zastavit. Čím vyšší je fitness funkce jedince, tím více políček o dané hodnotě zabírá.

Turnajová selekce

Při metodě turnajové selekce je náhodně vybráno N jedinců, kteří se účastní turnaje. Proměnná N může nabývat hodnot $N = 2$ až $N = N_{pop}$ (velikost populace), kdy se zvyšujícím se N roste i intenzita selekce. Pravděpodobnost postoupení do další generace je dána hodnotou fitness.

Reprodukce

Dalším krokem je generování další generace populace z těch, kteří jsou vybráni selekcí a následně podstupují operace křížení a mutaci. Tím, že je potomek vytvořen z rodičů z předchozí generace, sdílí s nimi mnoho společných charakteristik. Pro každého potomka jsou vybráni dva rodiče a tento proces se opakuje dokud není naplněna další generace. Do dalších generací postupují jedinci s vysokou fitness hodnotou, až se v poslední generaci opakuje téměř jediný nejlepší jedinec.

Křížení

Při použití binárního kódování je křížení operace, při které si rodiče vymění určité chromozomy a tím vzniknou dva potomci. U spojitého kódování s delším řetězcem $N_{var} > 2$ je křížení provedeno jako:

$$ma = \{p_{m1}, p_{m2}, \dots, p_{ma}, \dots, p_{mN_{var}}\}$$

$$da = \{p_{d1}, p_{d2}, \dots, p_{da}, \dots, p_{dN_{var}}\}$$

kde $p_{mN_{var}}$ a $p_{dN_{var}}$ jsou proměnné (chromozomy) pro daný prostor pro dvojici rodičů a p_{ma} a p_{da} je místo křížení. Celý proces křížení probíhá tak, že je náhodně vybráno místo křížení. V místě křížení (p_{ma} a p_{da}) je vygenerován nový parametr podle:

$$p_{new1} = p_{ma} - \beta[p_{ma} - p_{da}] \quad (4.2)$$

$$p_{new2} = p_{da} - \beta[p_{da} - p_{ma}] \quad (4.3)$$

kde proměnná β je náhodně vygenerované číslo v rozsahu $\beta \in < 0, 1 >$. Tímto jsou vytvořeny nové hodnoty, které nahradí hodnoty p_{ma} a p_{da} . Před hodnotou p_{new} zůstanou hodnoty prvního rodiče a za touto hodnotou pak parametry druhého rodiče. Potomci budou mít tvar:

$$potomek(1) = \{p_{m1}, p_{m2}, \dots, p_{new1}, \dots, p_{dN_{var}}\}$$

$$potomek(2) = \{p_{d1}, p_{d2}, \dots, p_{new2}, \dots, p_{mN_{var}}\}$$

Existuje více možností jak zkombinovat rodiče a získat různé potomky do nové generace.

Mutace

Mutace slouží k náhodné změně některých souřadnic jedinců. Počet mutací zvolíme pomocí vzorce 4.4:

$$mutace = \mu \times (N_{pop} - 1) \times N_{var} \quad (4.4)$$

kde μ je předem zvolená konstanta, např. $\mu = 0,2$. Následně jsou vygenerovány vektory pozic, na kterých dojde ke změně souřadnice. [1, 14, 20, 26]

4.2.3 Netopýří algoritmus

Netopýří algoritmus, jak již napovídá jeho název, je založen na chování echolokace netopýřů. Schopnost echolokace netopýřů je fascinující, protože netopýři jsou schopni najít svou kořist i rozeznat různé druhy hmyzu i v úplné tmě. Přestože každý impuls trvá několik tisícín sekundy, má konstantní frekvenci 25 kHz až 150 kHz. Při lovu může netopýř vyslat až 200 pulsů za sekundu, z čehož vyplývá obrovská schopnost zpracování signálu. Hlasitost takového signálu může dosahovat až 110 dB.

Pokud idealizujeme některé echolokační charakteristiky netopýřů, je možné je využít pro tyto algoritmy. Pro jednoduchost se uvádí tyto aproximace nebo idealizovaná pravidla:

1. Všichni netopýři používají echolokaci k zjištění vzdálenosti a také znají vzdálenost mezi potravou/kořistí a pozadím (bariérou).
2. Netopýři létají náhodně s rychlostí v_i na pozici x_i s konstantní frekvencí f_{min} , různou vlnovou délkou λ a hlasitostí A_0 ke hledání kořisti. Také mohou automaticky měnit vlnové délky (nebo frekvence) jejich vysílaného signálu v závislosti na druhu kořisti.
3. Také hlasitost se může měnit ve velkém rozsahu zahrnujícím velké pozitivní hlasitosti od A_0 až po minimální konstantu A_{min} .

Pohyb netopýra

Při simulacích se řídíme určitými pravidly, které určují rychlost v_i a pozici x_i daného netopýra v d-dimenzionálním prostoru.

$$f_i = f_{min} + (f_{max} - f_{min}) \cdot \beta \quad (4.5)$$

$$v_i^t = v_i^{t-1} + (x_i^t + x_*)_i \quad (4.6)$$

$$x_i^t = x_i^{t-1} + v_i^t \quad (4.7)$$

kde f je vyslaná minimální či maximální frekvence impulsů, v_i^t je rychlost i-tého netopýra v t-tém taktu, x_i^t je poloha i-tého netopýra v t-tém taktu a x_* je nejlepší poloha netopýra v populaci v daném momentu.

Náhodným výběrem se myslí porovnání náhodného čísla z rozsahu $< 0, 1 >$ s opakovací frekvencí r vysílaných impulsů. Hodnota r je normalizována do intervalu $< 0, 1 >$ a může se postupně zvyšovat k zadané hodnotě. Hodnota r je dána jako 4.8 :

$$r_i^{t+1} = r_i^0 \cdot (1 - e^{-\gamma t}) \quad (4.8)$$

kde γ je náhodné číslo z intervalu $< 0, 1 >$.

Oproti opakovací frekvenci se hodnoty hlasitosti vysílaných impulsů A mohou postupně snižovat (až k nule). Hodnota A je popsána vztahem 4.9 :

$$A_i^{t+1} = \alpha \cdot A_i^t \quad (4.9)$$

kde hodnoty náhodného čísla α jsou z intervalu $< 0, 1 >$. [2, 6]

5 IMPLEMENTACE METOD GLOBÁLNÍ OPTIMALIZACE

Na rozdíl od kapitoly 3, kde jsou použity funkce implementované v Matlabu, se budu v této kapitole zabývat funkcemi mnou vytvořenými. Uvedu zde stručný popis a ukázkou uživatelského rozhraní a jednotlivých funkcí popsanych v kapitole 4.

5.1 Vstupní parametry

Pro všechny vytvořené algoritmy jsou některé parametry shodné a některé jsou specifické jen pro daný algoritmus. Všechny potřebné vstupní parametry se zadávají do vytvořeného uživatelského rozhraní GUI.

Vstupní data

Vstupní data mohou být buď simulovaná nebo skutečná. Vstupní data jsou křivky, které jsou poté aproximovány zvoleným modelem a zvolenou optimalizační metodou.

Velikost populace

Vstupní parametr *velikostpopulace* udává, kolik jedinců bude rozmístěno po prohledávaném prostoru. Zde platí, že čím víc jedinců, tím hustší pokrytí budeme mít. Nevýhodou je, že se zvyšuje výpočetní náročnost a její doba. Velikost populace si může uživatel nastavit samostatně pro každý algoritmus. Většinou se ale volí stejně velké populace.

Počet iterací

Tato hodnota udává, kolikrát daný algoritmus proběhne. Její hodnoty se opět dají nastavit pro každý algoritmus zvlášť. Při velmi malé hodnotě nemusí program nutně dosáhnout globálního minima a tím i správného výsledku. Při opačném extrému je zase velmi dlouhá doba výpočtů. Proto volíme kompromis. Pro každý algoritmus je jiný optimální počet iterací.

Prohledávaný prostor

Pro nalezení hledaných parametrů rovnic je nutné omezit prohledávaný prostor. Při příliš velkém prohledávaném prostoru, by nemusel program najít správný výsledek. Tak i při příliš malém prostoru, by se mohly správné parametry přímek nacházet mimo něj. Tuto nevýhodu je možné omezit, protože námi prohledávaný

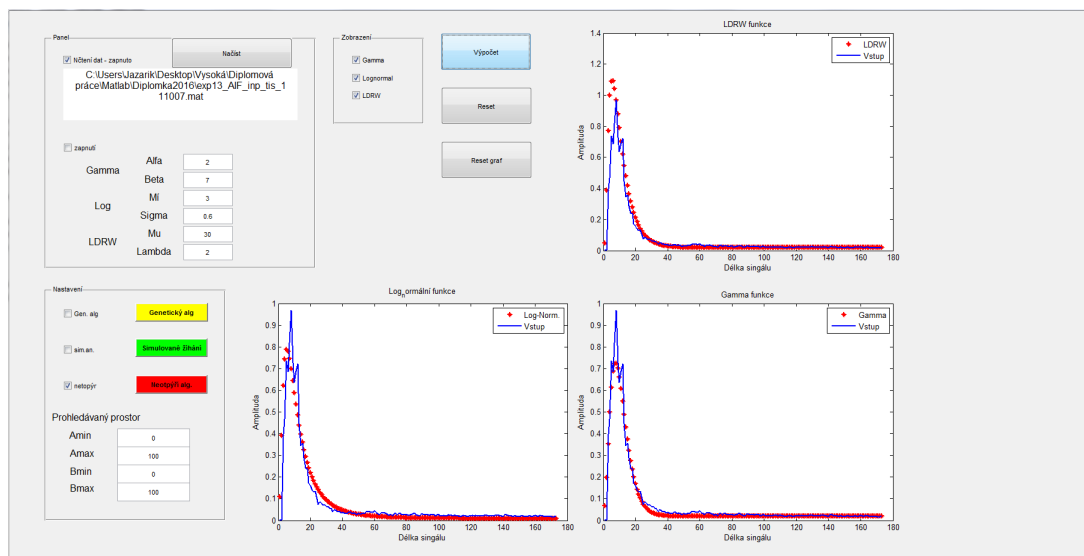
prostor se vždy zmenší na polovinu a celý program proběhne znovu. Tím se předejde nesprávnému výsledku. Do výstupu se vždy zapíše pouze nejlepší řešení z celého prohledávaného prostoru.

Koeficient exponentu

Protože modelované křivky málokdy odpovídají námi zadaným křivkám, je nutné přidat jednu proměnnou. Tou je v tomto případě koeficient exponentu k , kterým je umocněna proměnná t , která slouží jako časová základna. Poté je celá rovnice křivky vynásobena tímto členem. Při pokusech jsem odvodil určité hodnoty, které se hodí pro daný druh křivky, viz dále.

5.2 Popis uživatelského rozhraní

Pro přehlednější a jednodušší ovládání bylo vytvořeno uživatelské rozhraní GUI. Návrh předního panelu je zobrazen na 5.1.



Obr. 5.1: Uživatelské rozhraní GUI

Výběr dat

Při zapnutí programu se zobrazí uživatelské rozhraní. Zde v levém horním rohu lze zvolit zda budeme pracovat se simulovanými daty nebo se skutečnými. Při práci se skutečnými daty stačí jen zaškrtnout 'check box' *Načtení dat – zapnuto* a kliknout na pole *Načíst*, kde se dají vybrat požadovaná data. Všechny volané funkce si cestu k datům vyberou sami. Při práci se simulovanými daty stačí vybrat 'check box' pro simulovaná data, kde jsou již předvyplněné hodnoty.

Nastavení

V levém dolním rohu je panel *Nastavení*. V tomto oknu je možné si vybrat, kterou z globálních optimalizačních metod budou počítána výstupní data. Každou metodu lze vybrat zaškrtnutím 'check-box'. Pro každou metodu jsou již přednastavené jednotlivé parametry. Pokud by bylo nutné parametry změnit, lze to provést kliknutím na tlačítko dané metody. Zobrazí se okno s jednotlivými proměnnými. Poté stačí potvrdit nastavení. Dále je možné určit velikost prohledávaného prostoru nastavením hodnot (A_{min} , A_{max} , B_{min} , B_{max}). Pokud by byl prohledávaný prostor příliš veliký, nemusel by program najít správné hodnoty aproximace. Proto je do všech funkcí přidána podmínka, že horní hranice prohledávaného prostoru je vždy po skončení všech iterací zmenšena na polovinu. Tato podmínka platí do doby, dokud není horní hranice $A_{max} < 1$ a $B_{max} < 1$.

Poté si můžete zvolit pomocí kterých křivek budete aproximovat nalezené řešení. Pro vykreslení vámi zadaných hodnot a výstupů stačí stisknout tlačítko výpočet. Tlačítko Reset vymaže veškeré Vámi zadané nastavení. Tlačítko Reset graf vymaže vykreslené grafy.

Grafy

V uživatelském rozhraní jsou tři okna pro vykreslení výsledků. Každé okno zobrazuje jednu křivku, které aproximuje vypočítané hodnoty. Tedy do jednoho grafu se zobrazí všechny tři metody pro optimalizaci.

5.3 Funkce

Simulované žíhání

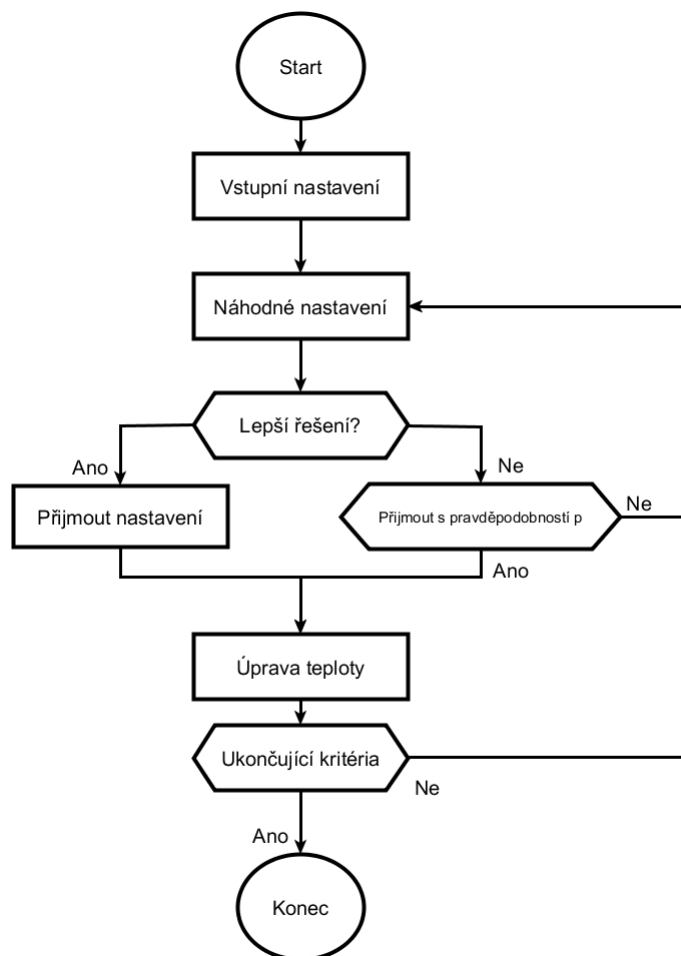
První optimalizační metodou je simulované žíhání (SA). Funkce pro skutečná data jsou v programu volána pomocí příkazu:

- `function [kvalita, populace, vystup,t,zaklad] =SA_data_gamma(fullpathname, vzorkovani_SA, poc_iter_SA, jedinci_SA, amin, amax, bmin, bmax, k_sa_gamma, poc_tep);`

Většina vstupních parametrů je popsána výše. Posledním parametrem je *poc_tep*, který zastupuje počáteční teplotu 'tělesa'.

Funkce pro modelová data jsou volána pomocí příkazů:

- `function [kvalita, populace, vystup, t,zaklad] = SA_gamma(alf, bet, vzorkovani_SA, poc_iter_SA, jedinci_SA, amin, amax, bmin, bmax);`



Obr. 5.2: Vývojový diagram simulovaného žíhání

Postup algoritmu je znázorněn na vývojovém diagramu 5.2. Vstupním nastavením se myslí zadání parametrů popsanych v kapitole 5.1. Náhodným nastavením se myslí generování počáteční populace a v průběhu programu i úpravu pozice jednotlivých jedinců. Poté se vypočítá funkční hodnota a porovná se s předešlou funkční hodnotou. Pokud je nová funkční hodnota lepší, pak je přijata. Pokud je horší, je tato hodnota přijata s určitou pravděpodobností (4.1). Tato pravděpodobnost je závislá právě na teplotě tělesa, která se zmenšuje v průběhu iterací. Program je ukončen, jestliže je dosaženo maximálního počtu iterací poc_iter_SA nebo je rozdíl funkčních hodnot jedince menší než zadaná přesnost. Dalšími podmínkami ukončení může být např. že se nejlepší jedinec opakuje v populaci v nadpoloviční většině, nebo je-li funkční hodnota jedinců v populaci stejná.

Genetický algoritmus

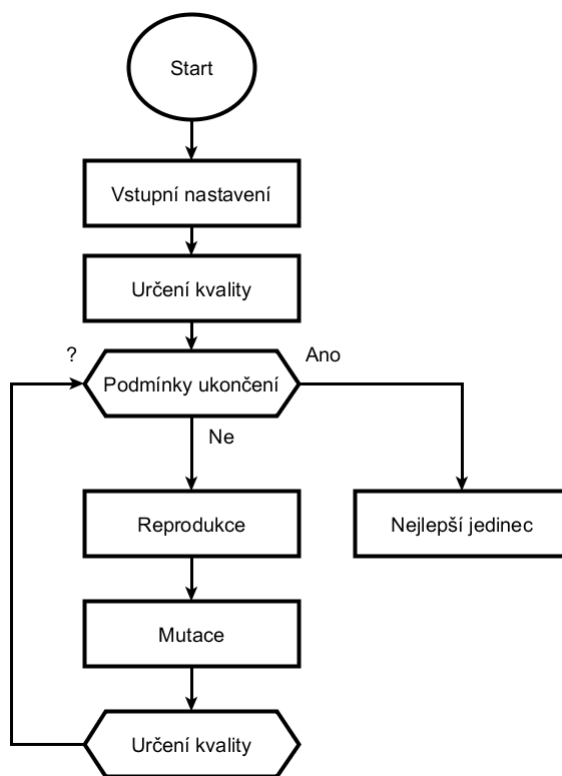
Mezi další použité funkce patří programy, využívající genetických algoritmů. Tyto funkce pro skutečná data jsou volány pomocí:

- `function [kvalita, populace, vystup,t,zaklad] =gen_data_gamma(fullpathname, vzorkovani_gen, poc_iter_gen, jedinci_gen, koef_mut_gen, amin, amax, bmin, bmax, k_gen_gamma);`

Jediným zatím nevysvětleným parametrem je vstupní proměnná *koef_mut_gen*, která určuje pravděpodobnost mutace při výpočtu. Tento program si také volá další dvě funkce. Jednou z nich je funkce *reprodukce*, která zajišťuje samotnou reprodukci a také křížení. Druhou funkcí je *mutace*, do které vstupuje již zmíněná proměnná. Funkce jsou volány pomocí:

- `populace=reprodukce(populace,kvalita,fcn,zaklad,AUC,t,delka);`
- `populace=mutace(populace,koef_mutace,jedinci);`

Algoritmus je zobrazen na vývojovém diagramu 5.3 Vstupní nastavení je zadání



Obr. 5.3: Vývojový diagram genetického algoritmu

koefficientů, nezbytných pro výpočet. Určení kvality odpovídá výpočtu funkční hodnoty pro danou funkci. Podmínkami ukončení je naplnění maximálního počtu iterací. Následují bloky reprodukce a mutace, které jsou popsány v 4.2. Následně je znovu

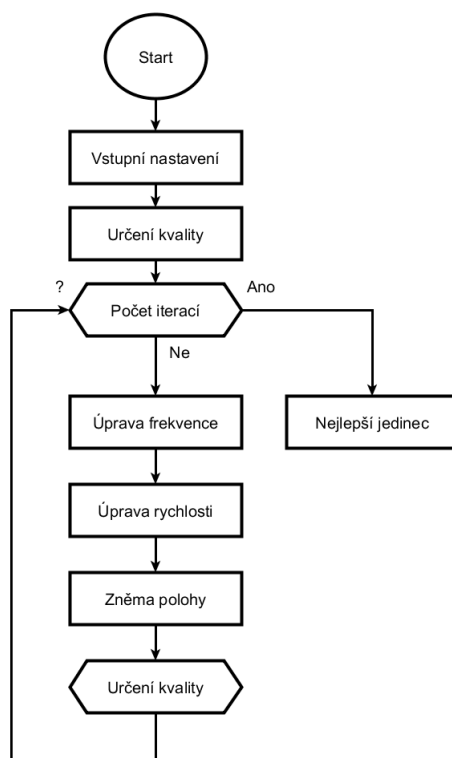
určena funkční hodnota a celý program se opakuje dokud není naplněn počet iterací. Výstupem jsou hodnoty nejlepšího jedince.

Netopýří algoritmus

Poslední volanou funkcí je netopýří algoritmus. Ten je volán pomocí:

- `function [kvalita, populace, vystup, t, zaklad] = netopyri_data_gamma(fullpathname, vzorkovani_net, jedinci_net, poc_iter_net, amin, amax, bmin, bmax, fmin, fmax, k_net_gamma);`

Při volání této funkce je použita proměnná, která je zmíněna v kapitole 4.2.3. Je jí frekvence $fmin$ a $fmax$, která vstupuje do dalších výpočtů. Postup algoritmu je zobrazen na 5.4.



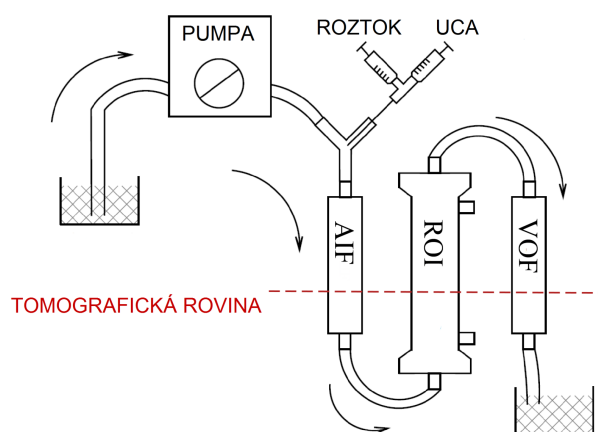
Obr. 5.4: Vývojový diagram netopýřího algoritmu

Bloky vstupní nastavení, určení kvality a počet iterací mají stejný význam jako u genetického algoritmu. Další bloky upravují samotný pohyb netopýra přes vzorce zmíněné v kapitole 4.2.3. Výstupem je nejlepší jedinec, jeho souřadnice a funkční hodnota.

6 TESTOVÁNÍ

6.1 Použitá data

Data pochází z fantomového experimentu provedeného v ultrazvukové laboratoři na UBMI. Schéma zapojení je na Obr. 6.1. Systém se skládá z peristaltické pumpy, systému propojovacích hadiček a dialyzačního filtru, který slouží jako model tkáně. Ultrazvuková RF data jsou snímána ve vyznačené rovině tak, aby v 2D obraze byly přítomny hadice modelující vstup a výstup ze systému a také dialyzační filtr. Měřený koncentrační signál je odvozený z obrazových hodnot v oblasti zájmu, kterou je oblast dialyzačního filtru. [3]



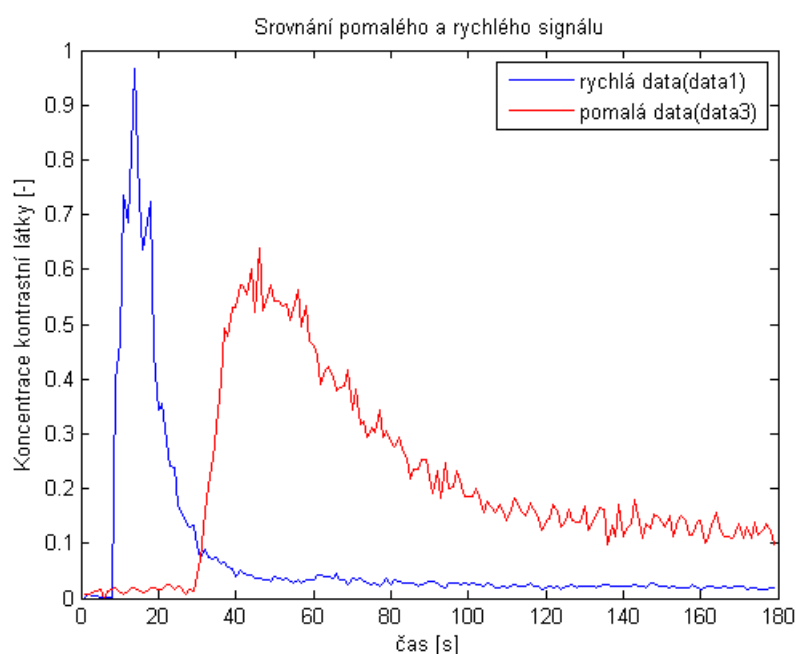
Obr. 6.1: Schématické zapojení fantomu

Data jsou upravená a uložena v proměnných *.mat*. Data se dají rozdělit na tzv. 'pomalá' a 'rychlá' data. Jejich rozdělení je v tabulce 6.1.

Ukázka rozdílu takových dat je zobrazena na grafu 6.2. Pro každý druh dat se hodí jiná aproximační funkce. Ukázka dat je upravená tak, aby maximální hodnota amplitudy byla rovna 1 a zároveň měli 'stejnou' nulovou linii.

Tab. 6.1: Tabulka testovaných dat

Rychlá data	exp13_AIF_inp_tis_111007.mat	data1
	exp13_aifnova_inp_tis_111017.mat	
	exp14_AIF2_inp_tis_111010.mat	
	exp14_AIF_inp_tis_110907.mat	data4
	exp15_AIF_inp_tis_111007.mat	
	exp17_AIF_inp_tis_111007.mat	
Pomalá data	exp13_roimale_inp_tis_111007.mat	data2
	exp13_roinove_inp_tis_111017.mat	
	exp13_roivelke_inp_tis_111007.mat	
	exp14_roimale2_inp_tis_111010.mat	
	exp14_roimale_inp_tis_110907.mat	
	exp14_roivelke_inp_tis_110907.mat	
	exp15_roimale_inp_tis_111007.mat	data3
	exp15_roivelke_inp_tis_111007.mat	
	exp17_roimale_inp_tis_111007.mat	
	exp17_roivelke_inp_tis_111007.mat	



Obr. 6.2: Srovnání pomalých a rychlých dat

Testovací nastavení

Zde uvedu pouze nastavení testovacího programu, při kterém bylo dosaženo výsledků popsanych v přílohách A.1, B.1, C.1, D.1. Výčet nastavení je zobrazen v G.1. Program automaticky mění velikost populace podle daného nastavení. Pro každé nastavení je provedeno 20 cyklů, aby bylo dosaženo validních výsledků. Počet iterací je nutné nastavit ručně, protože výpočetní čas pro počet iterací se velmi liší. V tabulce 6.2 jsou zobrazeny výpočetní časy jednotlivých algoritmů pro různý počet iterací. Časově nejméně náročný algoritmus je simulované žíhání. To je dáno náhodností celého procesu úpravy následné pozice každého jedince. Pro netopýří a genetický algoritmus se časová náročnost zvyšuje s násobkem iterací. Pro simulované žíhání to platí až od 1000 iterací. Ovšem při zvyšování počtu iterací první dvě metody neposkytují lepší výsledky, zatímco u simulovaného žíhání se pro úspěšnou aproximaci pohybuje počet iterací okolo 10^5 až 10^6 .

Tab. 6.2: Výpočetní náročnost algoritmu

Doba výpočtu	Počet iterací				
	10	100	1000	10000	100000
Netopýří algoritmus	0,448	3,124	30,112	313,474	-
Genetický algoritmus	1,089	9,989	99,556	996,837	-
Simulované žíhání	0,0817	0,0818	0,346	2,947	29,584

6.2 Výstupy

Množství výsledků na jednotlivých datových souborech je natolik velké, že není možné je všechny zahrnout do textu práce a jsou proto odevzdány na přiloženém CD ve formátu ".xlsx" souborech podle měřeného signálu. Dále jsou rozdělena podle počtu iterací a podle druhu výpočetní metody. Všechny zde uvedené funkční hodnoty jsou počítány podle 3.3 a vynásobeny hodnotou 10^4 , aby bylo možné lépe posoudit rozdíly mezi jednotlivými funkcemi a algoritmy.

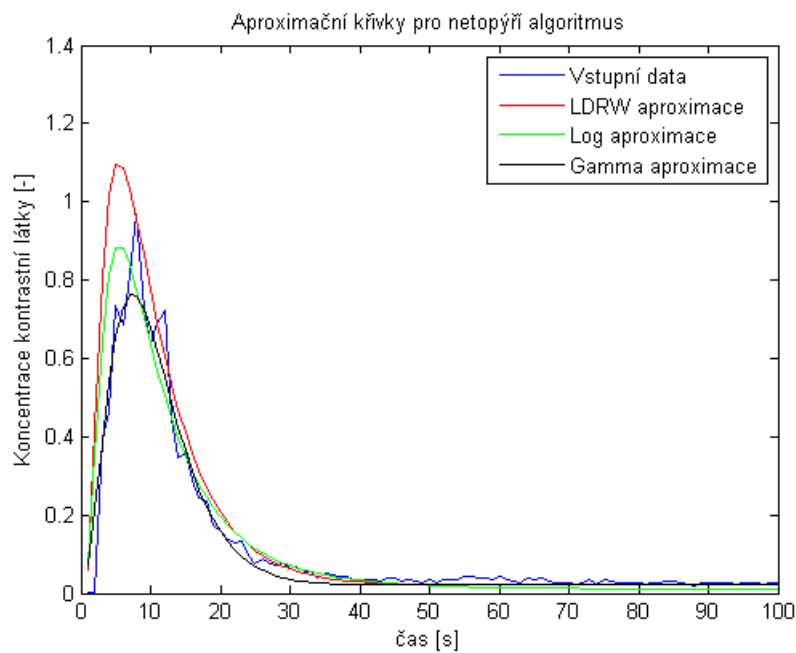
6.2.1 Výstupy pro data1

Veškeré naměřené hodnoty pro data1 jsou uložena v *Výstupní tabulka_exp13_AIF_inp_tis_111007.xlsx*. Nejlepší minimální funkční hodnoty a střední funkční hodnoty pro všechny kombinace aproximačních funkcí, algoritmů, počtu iterací a velikosti populace jsou zobrazeny v příloze A.1. Pro zjednodušení uvedu pouze nejlepší hodnoty pro každý algoritmus spolu s hodnotami proměnných. Ty jsou zobrazeny v tabulce 6.3.

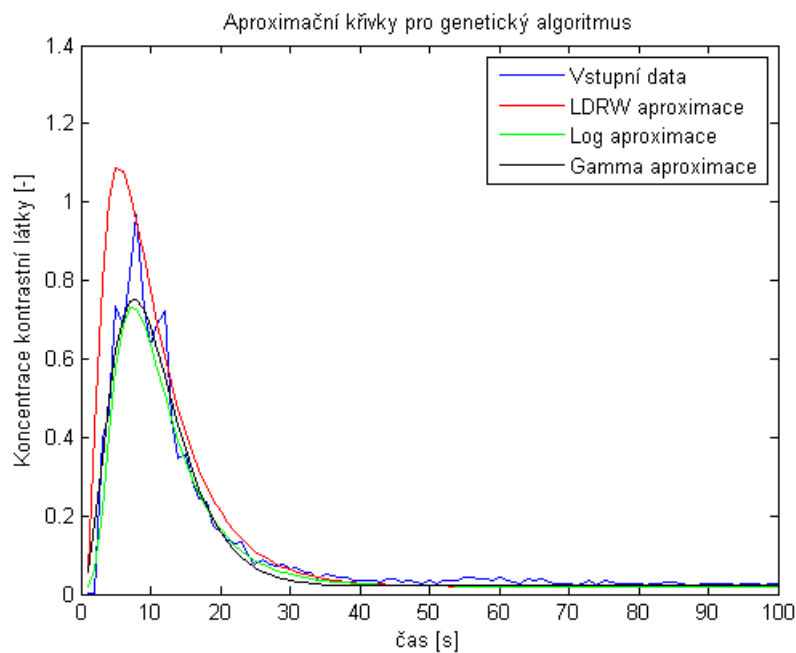
Tab. 6.3: Nejmenší funkční hodnoty pro jednotlivé kombinace

Algoritmy	Aproximační funkce		
	LDRW	Log-normální	Gamma
Netopýří	6,1740	5,5640	1,9630
Genetický	6,0280	1,8300	1,9850
Simulované žíhání	6,4920	5,8290	2,2500

Zobrazení aproximovaných dat pomocí jednotlivých modelů s využitím netopýřího algoritmu je na obr.6.3, s využitím genetického algoritmu na obr.6.4 a s využitím simulovaného žíhání na obr.6.5.

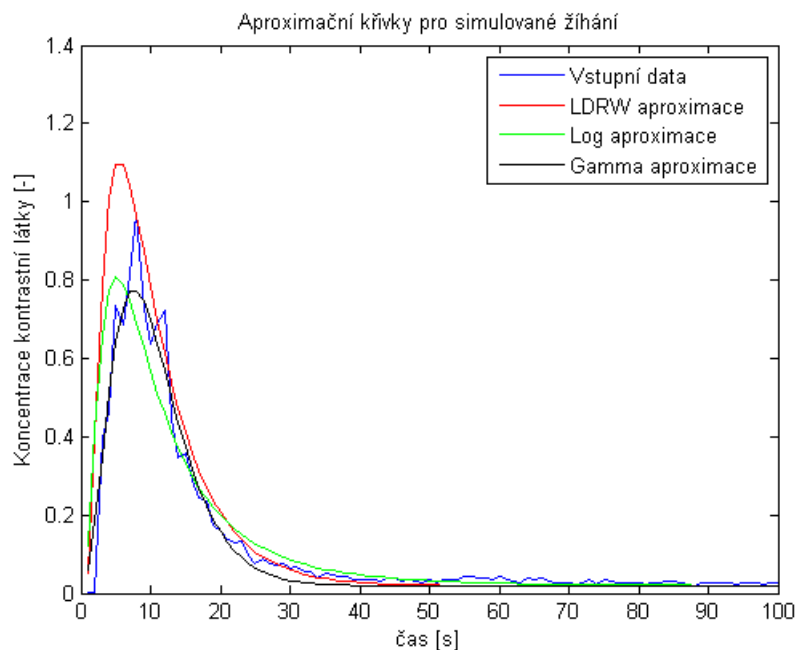


Obr. 6.3: Aproximace dat pomocí netopýřího algoritmu



Obr. 6.4: Aproximace dat pomocí genetického algoritmu

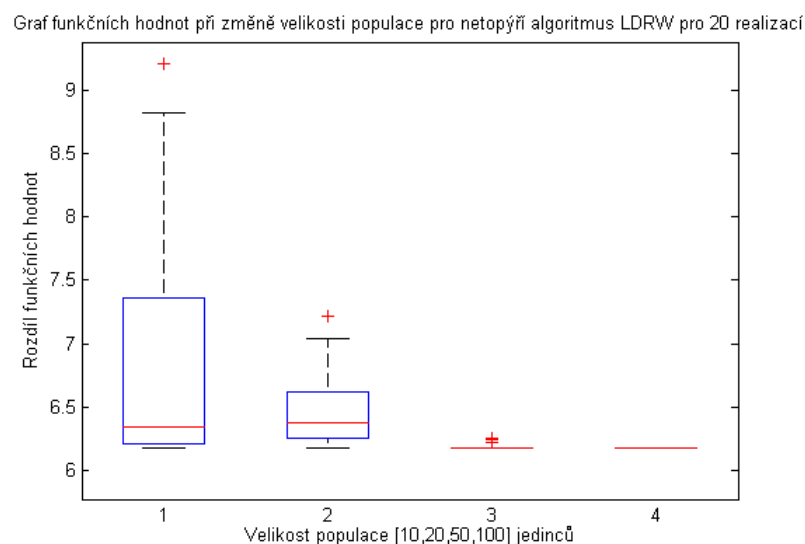
Z výstupních křivek je zřejmé, že nejlepší aproximaci u těchto rychlých dat poskytuje gamma funkce s využitím všech algoritmů. Pro genetický algoritmus stejný



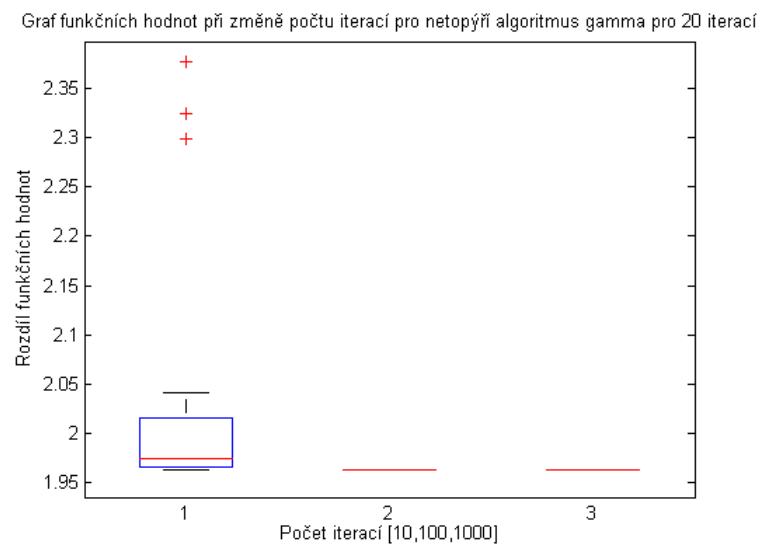
Obr. 6.5: Aproximace dat pomocí simulovaného žíhání

výsledek poskytuje i log-normální funkce.

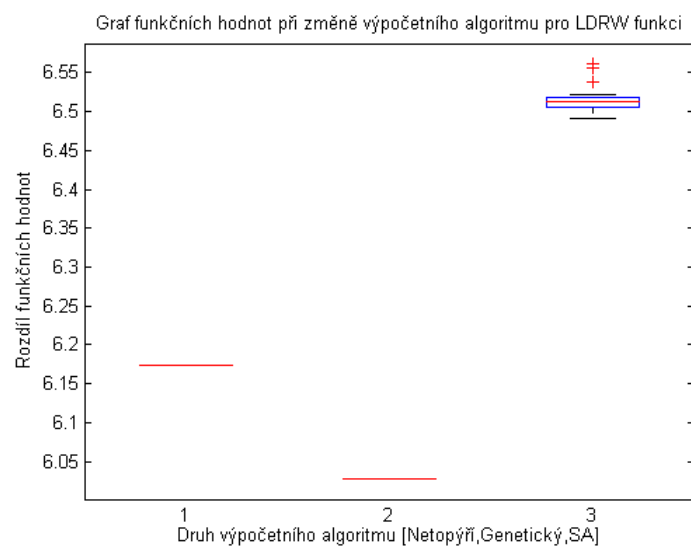
Pro lepší posouzení vlivu velikosti populace, iterací a druhu výpočetního algoritmu jsou vytvořeny grafy typu krabicový graf (boxplot). Tyto grafy jsou vytvořeny z dat aproximovaných pomocí LDRW funkcí.



Obr. 6.6: Srovnání funkčních hodnot pro změnu velikosti populace



Obr. 6.7: Srovnání funkčních hodnot pro změnu počtu iterací



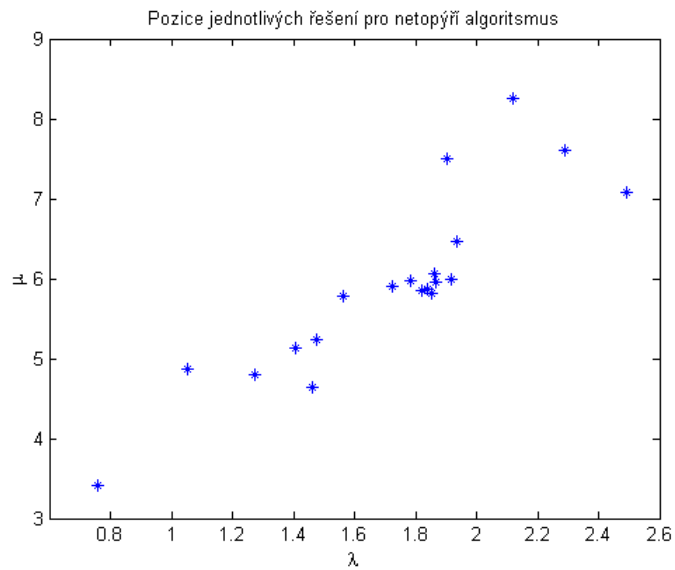
Obr. 6.8: Srovnání funkčních hodnot při změně výpočetního algoritmu

Na grafu 6.6 je vidět, že funkční hodnoty se zmenšují při zvětšování populace a rozptyl hodnot se zmenšuje. Až při populaci rovné 100 jedinců je výsledek neměnný. Téměř stejného výsledku je možné dosáhnout pro 50 jedinců. Tyto výsledky jsou pro netopýří algoritmus pro 1000 iterací.

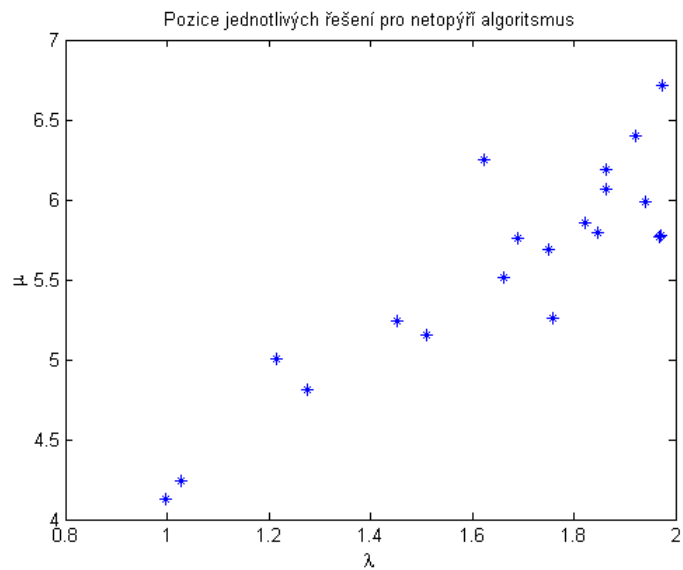
Při změně počtu iterací, které jsou zobrazeny na grafu 6.7 je zřejmé, že už při 100 iteracích je funkční hodnota shodná pro všech 20 opakování. Není zde žádný rozptyl ani odchylky. Minimální funkční hodnota je stejná pro všechny měření, ale střední hodnota a rozptyl se liší. Pro netopýří algoritmus je dostačující 100 iterací, po kterých je již dosažen nejlepší výsledek.

Při změně výpočetního algoritmu je z grafu 6.8 patrné, že nejlepší výsledky poskytuje genetický algoritmus. Genetický i netopýří algoritmus ovšem dosahují pro všechny opakování shodného výsledku. Zatímco u simulovaného žíhání nejsou všechny výstupní funkční hodnoty totožné. To může být způsobeno tím, že při používání simulovaného žíhání se pracuje s počtem iterací řádově 10^5 a více. Aby bylo možné porovnání s ostatními algoritmy, je použito maximálního počtu iterací 1000. Pro vyšší hodnoty by byly výpočetní doby pro genetický a netopýří algoritmus nepřiměřeně dlouhé a nebylo by dosaženo lepšího výsledku.

Z tabulky (6.3) tedy vyplývá, že nejlepší aproximaci pro *data1* je log-normální funkce vypočtená pomocí genetického algoritmu. Další je pak gamma funkce a nejhorší aproximaci poskytuje LDRW funkce pro všechny výpočetní metody.

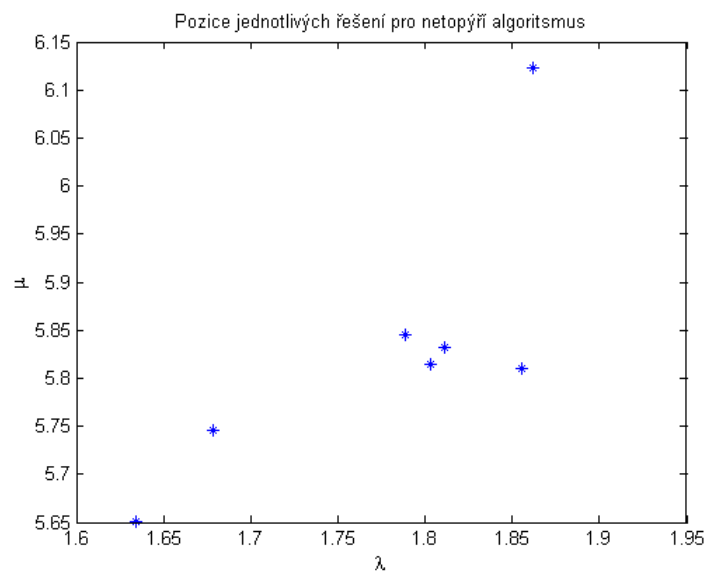


Obr. 6.9: Pozice parametrů pro LDRW funkci, velikost populace [10]

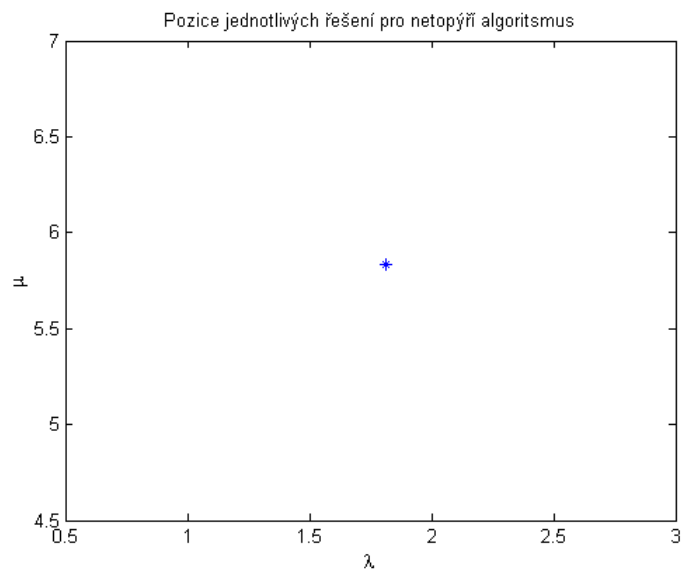


Obr. 6.10: Pozice parametrů pro LDRW funkci, velikost populace [20]

Grafy 6.9 , 6.10 ,6.11 ,6.12 zobrazují jednotlivé pozice koeficientů, které vstupují do LDRW funkce. Osa X odpovídá parametru λ a osa Y odpovídá parametru μ . Na grafu 6.9 jsou jednotlivá řešení rozmístěna ve velkém rozsahu. Na posledním grafu 6.12 je vidět pouze 1 bod. Pro velikost populace = 100 a počtu iterací =1000 konverguje program k jednomu výsledku pro všechny opakování.



Obr. 6.11: Pozice parametrů pro LDRW funkci, velikost populace [50]



Obr. 6.12: Pozice parametrů pro LDRW funkci, velikost populace [100]

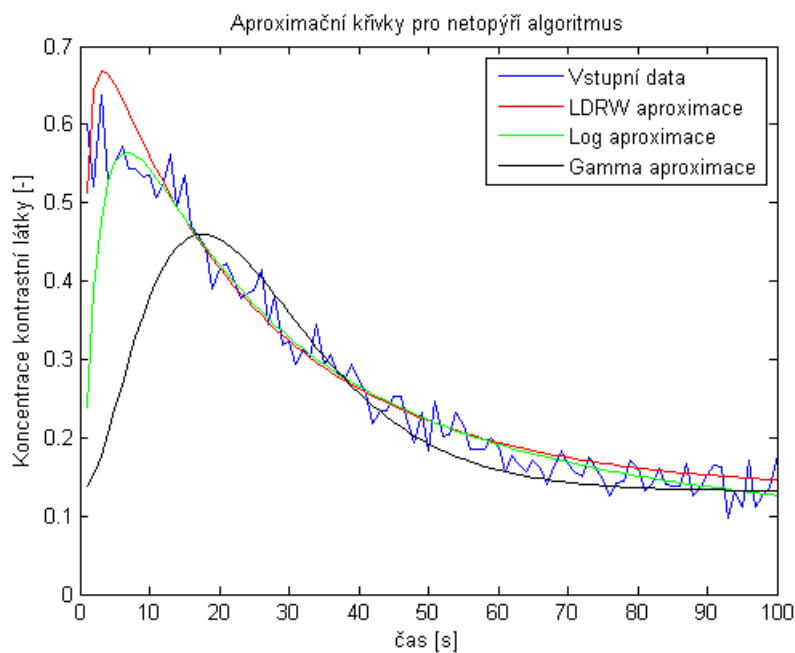
6.2.2 Výstupy pro data3

Jak již je zmíněno v tabulce 6.1, tak data3 patří do skupiny pomalých dat. Nejlepší výstupní funkční hodnoty pro tyto data jsou v 6.4. Z této tabulky lze vyvodit, že pro tyto data poskytuje nejlepší aproximaci funkce LDRW, následně Log-normální a nejhorší je gamma funkce.

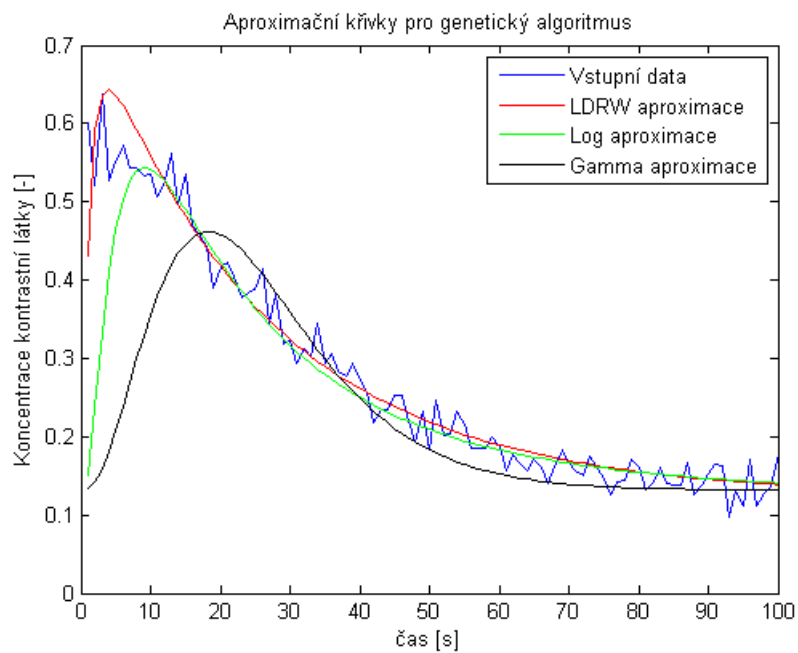
Tab. 6.4: Nejmenší funkční hodnoty pro jednotlivé kombinace

Algoritmy	Aproximační funkce		
	LDRW	Log-normální	Gamma
Netopýří	4,525	4,273	19,418
Genetický	3,737	6,145	21,935
Simulované žíhání	3,74	5,98	25,214

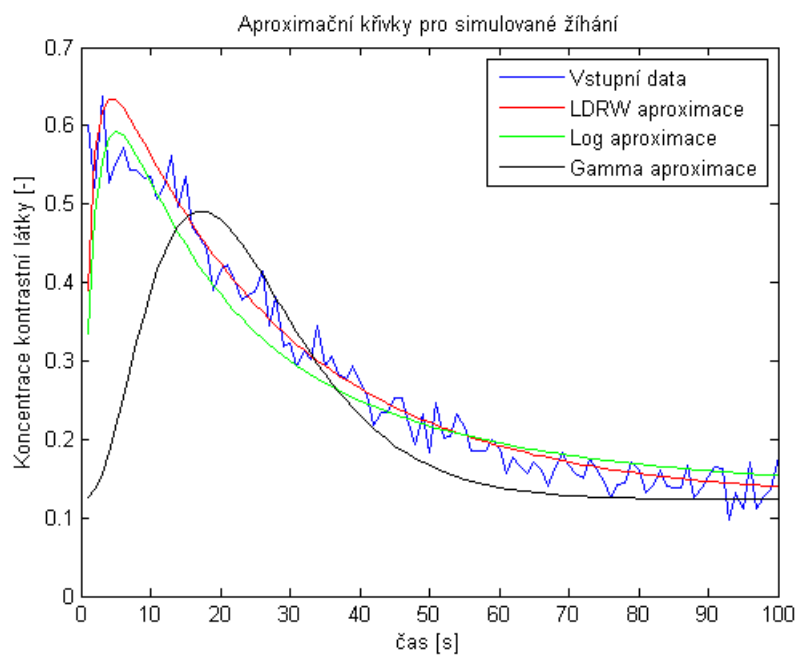
Výstupy aproximačních funkcí jsou zobrazeny na grafech 6.13, 6.14, 6.15.



Obr. 6.13: Aproximace dat pomocí netopýřího algoritmu



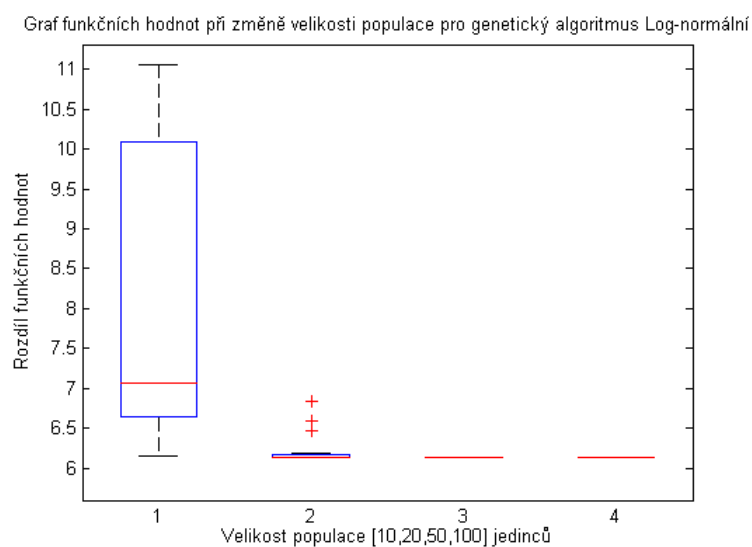
Obr. 6.14: Aproximace dat pomocí genetického algoritmu



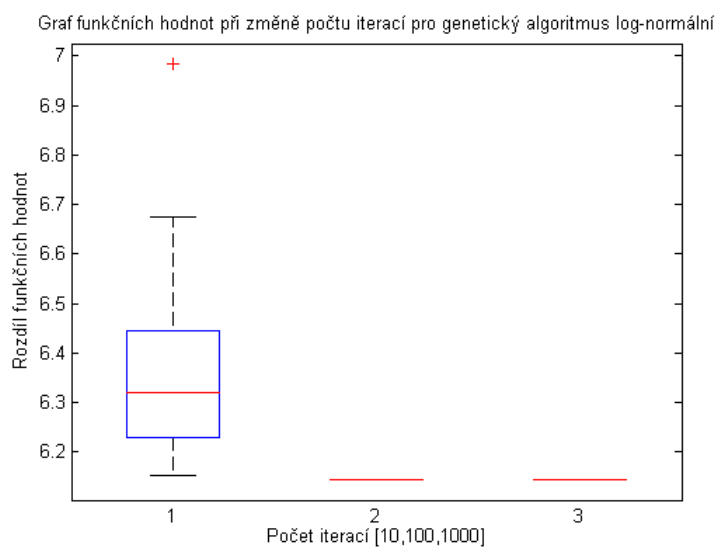
Obr. 6.15: Aproximace dat pomocí simulovaného žíhání

Pro data3 se jako nejlepší aproximační funkce jeví LDRW proložení. Následně log-normální a nejhorší aproximaci poskytuje gamma funkce.

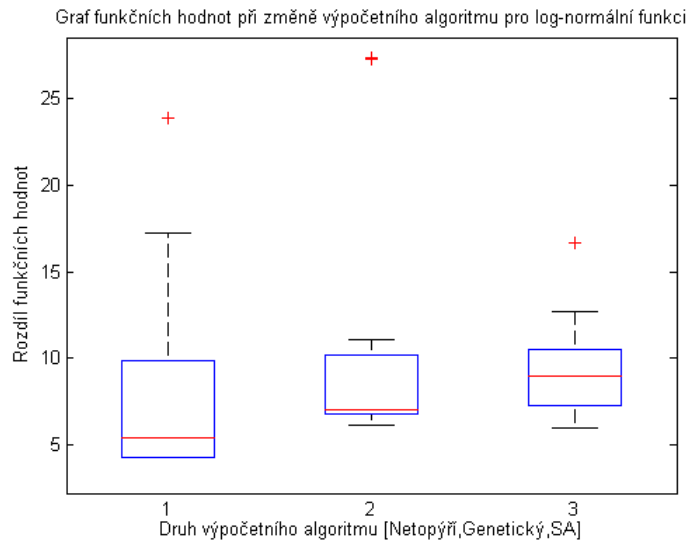
Grafy 6.16 , 6.17 , 6.18 ukazují změny výsledných funkčních hodnot pro 20 opakování.



Obr. 6.16: Srovnání funkčních hodnot pro změnu velikosti populace



Obr. 6.17: Srovnání funkčních hodnot pro změnu počtu iterací



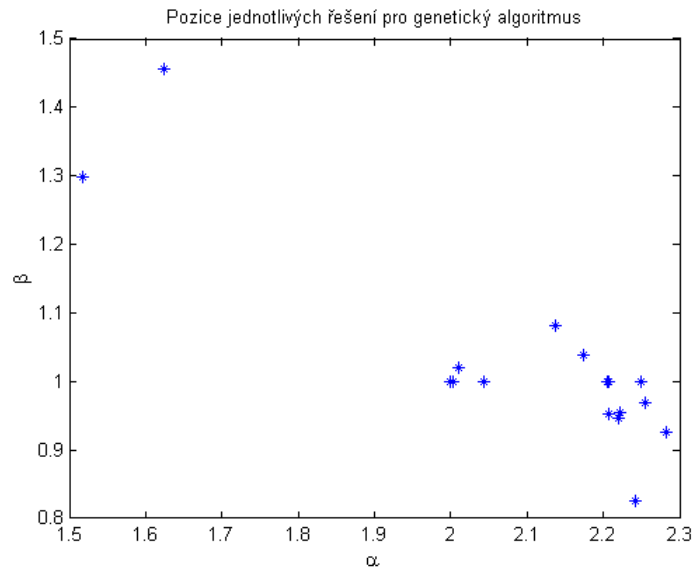
Obr. 6.18: Srovnání funkčních hodnot při změně výpočetního algoritmu

Z grafu 6.16 je zřejmé, že výsledná funkční hodnota je shodná pro velikost populace $n=[50,100]$. Při velikosti populace $[20]$ jsou výchyly minimální. Pro populaci rovnou $n=[10]$ jsou výsledky funkční hodnoty rozmístěny na velkém intervalu a nemusí poskytnout relevantní výsledek.

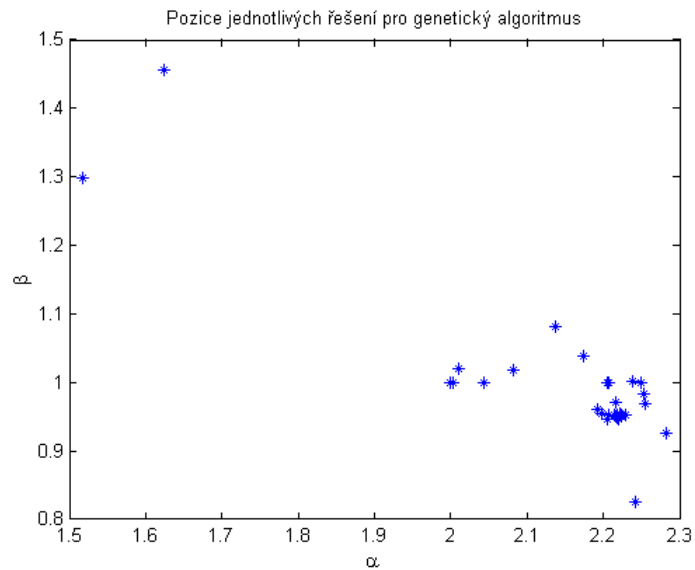
Pro genetický algoritmus je při počtu iterací $[100 \text{ a } 1000]$ funkční hodnota pro všechny opakování shodná. Správného výsledku je tedy možno dosáhnout i při menším počtu iterací a s nižší časovou náročností.

Graf 6.18 je zobrazen pro jednotlivé výpočetní algoritmy pro 100 iterací a velikost populace $n=10$. Nejmenší funkční hodnotu a tím i nejlepší řešení poskytuje netopýří algoritmus. Pro každý algoritmus lze naléznout výstupní koeficienty, které by při dosazení neposkytovali aproximaci odpovídající vstupním datům. Tyto koeficienty jsou v grafu zobrazeny jako červené křížky, které vybočují z určitého rozsahu. Z tohoto grafu také vyplývá důležitost počtu iterací při výpočtu.

Jednotlivá řešení pro genetický algoritmus při aproximaci pomocí log-normální funkce jsou zobrazena na grafech 6.19 , 6.20 , 6.21 , 6.22.

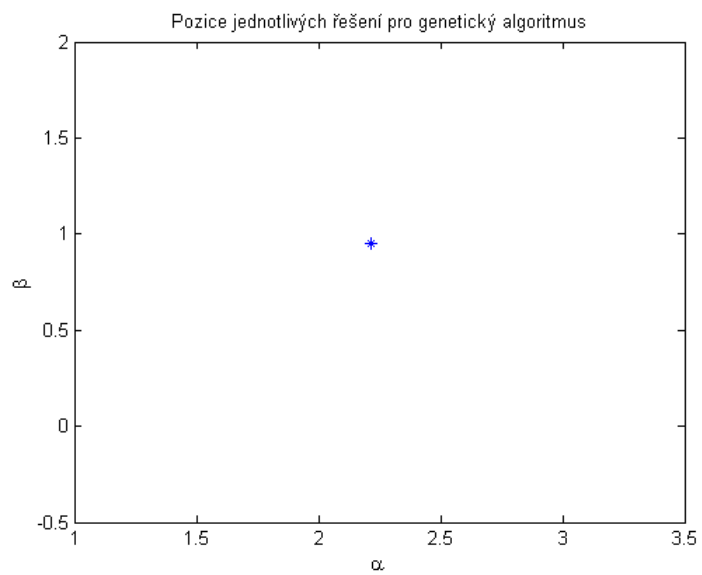


Obr. 6.19: Pozice parametrů pro log-normální funkci, velikost populace [10]

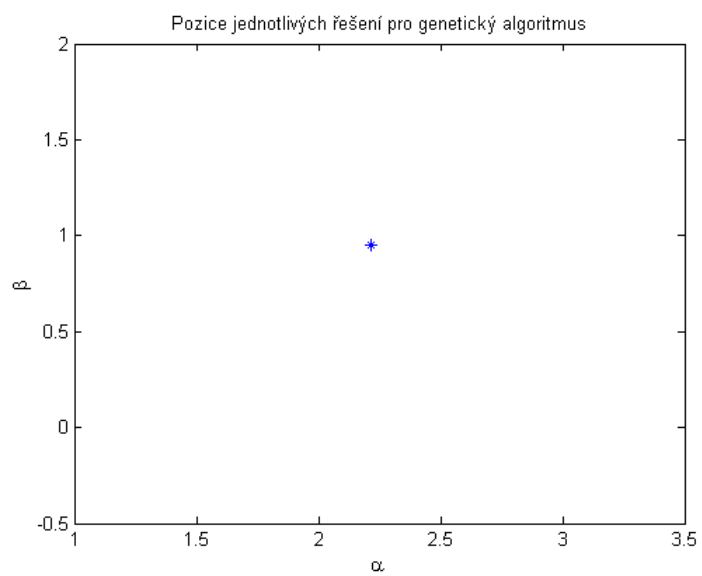


Obr. 6.20: Pozice parametrů pro log-normální funkci, velikost populace [20]

Rozdíl při změně velikosti populace je nejlépe znázorněn na grafech 6.20 a 6.21 kde je vidět, že při větším pokrytí prohledávaného prostoru program vždy najde 'správné' řešení. Výsledkem je tedy 1 bod, kdy jsou všechny hodnoty parametrů pro aproximaci shodné. Z grafu 6.22 je vidět, že ani velikost populace $n=100$ neposkytuje lepší výsledek než při $n=50$.



Obr. 6.21: Pozice parametrů pro log-normální funkci, velikost populace [50]



Obr. 6.22: Pozice parametrů pro log-normální funkci, velikost populace [100]

6.2.3 Výstupy pro data2

Pro data2 zde uvedu pouze výstupní tabulku s minimálními funkčními hodnotami, tabulka 6.5. Ostatní tabulky a grafy jsou uvedeny v přílohách B.1 a E.

Tab. 6.5: Nejmenší funkční hodnoty pro jednotlivé kombinace

Aproximační funkce			
Algoritmy	LDRW	Log-normální	Gamma
Netopýří	4,6098	7,1451	13,7207
Genetický	4,6384	6,8751	16,4632
Simulované žíhání	4,6517	9,7468	16,7111

6.2.4 Výstupy pro data4

Tabulka nejlepších funkčních hodnot pro všechny kombinace je uvedena v tabulce 6.6. Grafické výstupy jsou uvedeny v příloze F a tabulky v příloze D.1.

Tab. 6.6: Nejmenší funkční hodnoty pro jednotlivé kombinace

Aproximační funkce			
Algoritmy	LDRW	Log-normální	Gamma
Netopýří	9,315	7,155	4,261
Genetický	9,682	3,105	4,370
Simulované žíhání	9,684	3,730	4,576

7 DISKUZE

V této práci jsem se zabýval globální optimalizací a testováním algoritmů pro aproximaci dat pomocí funkcí. Výstupem je funkční program pro aproximaci dat a jejich zobrazení. V první části jsou použity modelové křivky a jejich proložení pomocí funkcí implementovaných v programu Matlab. Výstupem první části je odzkoušení jednotlivých aproximačních funkcí vůči jednotlivým modelovým datům. Tedy vstupem může být např. LDRW funkce a na její proložení jsou použity ostatní funkce. Abychom dosáhli určitého výsledku, jsou vstupní data zatížena šumem.

Z testování vyplývá, že pro rychlá data se jako aproximační křivka hodí gamma funkce, následovaná log-normální funkcí. Nejhorší výsledky poskytuje LDRW funkce. Z porovnání průběhů gamma a LDRW distribuce je patrné, že LDRW funkce má rychlou náběžnou hranu, ale pomalejší sestup za vrcholem. Toho je využito u pomalých dat. Gamma distribuce má také rychlou náběžnou hranu, ale zároveň má i rychlý sestup. Z testovaných dat je možné pozorovat, že log-normální funkci je možno použít pro oba druhy dat.

Vliv velikosti populace je zřejmý jak z krabicových grafů, tak z výstupních tabulek. Tento parametr určuje pokrytí prohledávaného prostoru jedinci. Při malém počtu jedinců nemusí program dosáhnout správného výsledku. S velikostí populace roste ovšem i výpočetní náročnost programu.

Počet iterací jednotlivých programů ovlivňuje dosažený výsledek. Z testování vyšlo, že pro genetický a netopýří algoritmus je maximální počet iterací roven 1000, zatímco pro simulované žíhání je 1000 iterací nejnižší hranicí, při které jsou výsledky téměř stejné, jen s malým rozptylem. U simulovaného žíhání je nutné používat tedy velkého počtu iterací, řádově 10^5 , protože tento algoritmus využívá náhodné změny pozice daného jedince. Ostatní programy využívají cíleného prohledávání v okolí průběžně nejlepšího jedince.

Časová náročnost programu roste s počtem iterací, velikostí populace a velikostí prohledávaného prostoru. To je dáno tím, že všechny programy pracují v cyklech, které jsou vnořené.

Výstupem práce jsou tedy funkce, které aproximují křivky naměřené při použití echokontrastní ultra-sonografie. Použité modely se již v praxi používají. Např. v článku [9], "Comparison of four mathematical models to analyze indicator-dilution curves in the coronary circulation" jsou použity tyto modely pro analýzu srdečního výdeje či měření průtoku krve. Další ukázkou lze najít v článku [8] "Comparison and evaluation of indicator dilution models for bolus of ultrasound contrast agents", kde jsou použity jednotlivé funkce pro proložení modelových dat.

8 ZÁVĚR

V úvodu diplomové práce jsou uvedeny základní informace o ultrazvuku a základy lékařského zobrazení. Dále jsou popsány kontrastní látky a perfuzní zobrazení. V následující kapitole jsou uvedeny matematické modely, farmakokinetické modely a modely pro perfuzní zobrazení. V praktické části je uveden vznik signálu a modelového šumu, popis funkcí a optimalizace signálu pomocí Optimization toolboxu, aby výsledný signál odpovídal originálnímu signálu.

V následujících kapitolách jsem se zabýval metodami globální optimalizace. Jsou zde rozebrány teoretické vztahy a popis jednotlivých modelů. Prvním modelem je simulované žíhání, následně je popsán genetický algoritmus a poslední je netopýří algoritmus.

Práce obsahuje uživatelské rozhraní vytvořené v prostředí Matlab tak, aby bylo co nejjednodušší pro koncového uživatele. Uživatel si zde může nastavit jednotlivé parametry, funkce a algoritmy, pomocí kterých budou vstupní křivky aproximovány. Další kapitola se zabývá praktickým testováním jednotlivých programů. Program byl testován na 4 různých datech, získaných z fantomového experimentu na UBMI. Pro tyto data jsou výsledky uvedeny jak v textu ve zkrácené podobě tak v příloze. Testovaná data jsou rozdělena na pomalá a rychlá. Pro aproximaci rychlých dat modelem se nejlépe hodí gamma funkce. Pro pomalá data je výhodnější LDRW funkce. V celé práci pracuji s různým nastavením počáteční populace a počtu iterací. Pro genetický a netopýří algoritmus je dostačují 100 až 1000 iterací. Pro simulované žíhání je vhodnější použít většího počtu iterací, protože výpočetní doba nedosahuje náročnosti předchozích metod a poskytuje lepší výsledky. Obecně platí, že vhodnější je použít větší počáteční populaci, kvůli pokrytí prohledávaného prostoru.

Díky této diplomové práci bylo ověřeno, že lze využít metody globální optimalizace ze skupiny evolučních algoritmů pro úlohu minimalizace střední kvadratické chyby u proložení dat vhodným modelem. Rozšířením této práce by mohl být výpočet jednotlivých parametrů vstupních křivek a jejich srovnání vzhledem k aproximační funkci.

LITERATURA

- [1] *A Continuous Genetic Algorithm Designed for the Global Optimization of Multimodal Functions*. CHELOUAH, Rachid a Patrick SIARRY. *Journal of Heuristics* [online]. 6. Kluwer Academic Publishers, 2016, s. 191-213 [cit. 2016-05-14].
- [2] XIN-SHE, Yang. *A New Metaheuristic Bat-Inspired Algorithm: in: Nature Inspired Cooperative Strategies for Optimization*. Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, UK: NISCO 2010, 2010, 65-74.
- [3] MÉZL, M.; JIŘÍK, R.; HARABIŠ, V.; KOLÁŘ, R.; STANDARA, M.; NYLUND, K.; GILJA, O.; TAXT, T.: *Absolute ultrasound perfusion parameter quantification of a tissue-mimicking phantom using bolus tracking.. IEEE Transactions on Ultrasonocs, Ferroelectrics, and Frequency Control*, 2015, roč. 62, č. 5, s. 983-987. ISSN: 0885- 3010.
- [4] CAROVAC, Aladin, Fahrudin SMAJLOVIC a Dzelaludin JUNUZOVIC. *Application of Ultrasound in Medicine*. *Acta Informatica Medica* [online]. 2011, : 168- [cit. 2015-10-25]. DOI: 10.5455/aim.2011.19.168-171. ISSN 0353-8109. Dostupné z: <http://www.scopemed.org/?mno=11306>
- [5] STØYLEN, Asbjørn. *Basic ultrasound, echocardiography and Doppler for clinicians*. In: *Norwegian University of Science and Technology* [online]. 2015, 2015-1-1 [cit. 2015-10-24]. Dostupné z: <http://folk.ntnu.no/stoylen/strainrate/Ultrasound/>
- [6] Yang, X. S., (2011). *Bat Algorithm for Multiobjective Optimization*, *Int. J. Bio-Inspired Computation*, Vol.3, No.5, pp.267-274
- [7] HRAZDIRA, Ivo. *Biofyzikální základy ultrasonografie: jak pracovat s ultrazvukovým diagnostickým přístrojem : praktická příručka s teoretickým úvodem pro stáž připravenou v rámci projektu: "Prohloubení odborné spolupráce a propojení ústavů lékařské biofyziky na lékařských fakultách v České republice"*. HRAZDIRA, Ivo. [online]. 1. vyd. Olomouc: Univerzita Palackého v Olomouci, 2011, s. 70 [cit. 2015-10-25]. DOI: ISBN 978-80-244-2895-6. ISBN 978-80-244-2895-6.
- [8] HARABIS, Vratislav, Radim KOLAR, Martin MEZL a Radovan JIŘÍK. *Comparison and evaluation of indicator dilution models for bolus of ultrasound contrast agents*. In: *Physiological Measurement* [online]. 2013, 34(2), s. 151-162 [cit. 2016-05-17]. DOI: 10.1088/0967-3334/34/2/151. ISSN 0967-3334. Dostupné z: <http://stacks.iop.org/0967-3334/34/i=2/a=151?key=crossref.53e8f12ae214d441c5ed0f32c86c6168>

- [9] BRANDS, Judith, Hans VINK a Jurgen W. G. E. VAN TEEFFELEN. *Comparison of four mathematical models to analyze indicator-dilution curves in the coronary circulation*. In: Medical [online]. 2011, 49(12), s. 1471-1479 [cit. 2016-05-17]. DOI: 10.1007/s11517-011-0845-9. ISSN 0140-0118. Dostupné z: <http://link.springer.com/10.1007/s11517-011-0845-9>
- [10] THE MATHWORKS, INC. *Documentation: lsqcurvefit* [online]. United States, 1994, 2015 [cit. 2015-12-27]. Dostupné z: <http://www.mathworks.com/help/optim/ug/lsqcurvefit.html>
- [11] UNGERMANN, Leoš, Pavel ELIÁŠ, Pavel RYŠKA, Antoním MICHL, Jan ŽIŽKA a Luděk KLZO. *Dynamická kontrastní ultrasonografie jater: Contrast-enhanced ultrasound of liver diseases. Česká radiologie: Czech Radiology : Časopis Radiologické společnosti* [online]. 1. Praha: Nakladatelské a tiskové středisko ČLS JEP, 1994-, 28.9.2009, 63(1): 34-41 [cit. 2015-10-31]. ISSN 1210-7883. Dostupné z: http://www.cesradiol.cz/dwnld/Ces_Rad_0901_34_41.pdf
- [12] NANDA, Navin C. *History of echocardiographic contrast agents. Clinical Cardiology* [online]. 1997, 20(S1): 7-11 [cit. 2015-10-31]. DOI: 10.1002/clc.4960201304. ISSN 01609289. Dostupné z: <http://doi.wiley.com/10.1002/clc.4960201304>
- [13] STROUTHOS, Costas, Marios LAMPASKIS, Vassilis SBOROS, Alan MCNEILLY a Michalakis AVERKIOU. *Indicator dilution models for the quantification of microvascular blood flow with bolus administration of ultrasound contrast agents. IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control* [online]. 2010, 57(6): 1296-1310 [cit. 2015-10-31]. DOI: 10.1109/TUFFC.2010.1550. ISSN 0885-3010. Dostupné z: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5480173>
- [14] *International journal of scientific: Multiobjective Programming With Continuous Genetic Algorithm* [online]. India: International Journal of Scientific, 2012 [cit. 2016-03-18]. ISBN 2277-8616. Dostupné z: <http://www.ijstr.org/final-print/sep2014/Multiobjective-Programming-With-Continuous-Genetic-Algorithm.pdf>
- [15] PAVLÍK, Tomáš. Logaritmicko-normální rozdělení. In: INSTITUT BIOSTATISTIKY A ANALÝZ MASARYKOVY UNIVERZITY. *Logaritmicko-normální rozdělení* [online]. Brno, 2015 [cit. 2015-11-30]. Dostupné z: <http://portal.matematickabiologie.cz/index.phppgāplikovanaanalyza-klinickychabiologickychdat-biostatistikapromatematickoubiologii-nahodnavelicinarozdelenipravdepodobnostiarealnadata-dalsirozdeleni-pravdepodobnosti-logaritmickonormalnirozdeleni>

- [16] NAVRÁTIL, Leoš a Jozef ROSINA. *Medicínská biofyzika*. Vyd. 1. Praha: Grada, 2005, 524 s. ISBN 80-247-1152-4.
- [17] DEJONG, N. *Mechanical Index. European Journal of Echocardiography: Mechanical Index* [online]. 2002, 73-74 [cit. 2015-10-25]. DOI: 10.1053/euje.2001.0138. ISSN 15252167. Dostupné z: <http://ehjcmimaging.oxfordjournals.org/cgi/doi/10.1053/euje.2001.0138>
- [18] JIŘÍK, Radovan. *Modely v biologii a epidemiologii* [online]. Brno, 2006, 68 s. [cit. 2015-11-01].
- [19] *Modes of ultrasound. Critical USG: Ultrasound in anesthesiology and intensive care* [online]. 2012, 2012 [cit. 2015-10-24]. Dostupné z: <http://www.criticalusg.pl/en/physical-principles/modes-of-ultrasound>
- [20] FITA, Adunga. *Multiobjective Programming With Continuous Genetic Algorithm. INTERNATIONAL JOURNAL OF SCIENTIFIC TECHNOLOGY RESEARCH*. 2014, 2014(Volume 3), 135-149. ISSN 227-8616.
- [21] TONG, K.H., Norhisham BAKHARY, A.B.H. KUEH a A.Y. Mohd YASSIN. *Optimal sensor placement for mode shapes using improved simulated annealing. Smart Structures and Systems* [online]. 2014, 13(3), s. 389-406 [cit. 2016-03-07]. DOI: 10.12989/sss.2014.13.3.389. ISSN 1738-1584. Dostupné z: <http://koreascience.or.kr/journal/view.jsp?kj=KJKHFZ>
- [22] *Optimization by Simulated Annealing. KIRKPATRICK, S. Science: New Series* [online]. 220. 1983, s. 671-680 [cit. 2016-05-12].
- [23] HARABIŠ, Vratislav. *Registrace obrazů - aplikace v oftalmologii a ultrasonografii* [online]. Brno, 2013, 109 s. [cit. 2015-10-24]. Dostupné z: https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.phpfile_id=70051. Dizertační práce. Vysoké učení technické v Brně. Vedoucí práce Doc. Ing. RADIM KOLÁR, Ph.D.
- [24] *Simulated Annealing. BERTSIMAS, Dimitris a John TSITSIKLIS. Statistical Science* [online]. 8. 1993, s. 10-15 [cit. 2016-05-14].
- [25] MORGAN, Matt A. a Mirjan M. NADRJANSKY. *Ultrasound frequencies*. Radiopaedia.org [online]. 2015 [cit. 2015-10-24]. Dostupné z: <http://radiopaedia.org/articles/ultrasound-frequencies>
- [26] ZELINKA, Ivan. *Umělá inteligence: v problémech globální optimalizace*. 1. vyd. Praha: BEN - technická literatura, 2002. ISBN 80-730-0069-5.

- [27] HŘEBÍČEK, Jiří a Michal ŠKRDLA. *Úvod do matematického modelování* [online]. 2006 [cit. 2015-11-01]. Dostupné z: <https://is.muni.cz/el/1431/podzim2007/Bi3101/um/skripta.pdf>
- [28] HRAZDÍRA, Ivo. *Úvod do ultrasonografie: V otázkách a odpovědích pro studenty lékařské fakulty* [online]. Brno, 2008 [cit. 2015-10-24]. Dostupné z: http://www.med.muni.cz/dokumenty/pdf/uvod_o_ultrasonografii1.pdf

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

CW	Continuous wave
PW	Pulse wave
CEUS	Contrast enhanced ultrasound
SAE	Stimulovaná akustická emise
AUC	Area under the time-intensity curve
MTT	Mean transit time
MRT	Mean residence time
LDRW	Local density random walk

SEZNAM PŘÍLOH

A	Tabulka výsledků pro data1	71
B	Tabulka výsledků pro data2	72
C	Tabulka výsledků pro data3	73
D	Tabulka výsledků pro data4	74
E	Grafy výsledků pro data2	75
F	Grafy výsledků pro data4	78
G	Druhá příloha	81
H	Obsah přiloženého CD	82

A TABULKA VÝSLEDKŮ PRO DATA1

Tab. A.1: Tabulka hodnot pro data exp13_AIF_inp_tis_111007.mat

Minimální funkční hodnoty	Iterace	100				1000							
	Populace	10	20	50	100	10	20	50	100				
Netopýří algoritmus	LDRW	6,189	6,196	6,180	6,174	6,184	6,187	6,174	6,174	6,176	6,180	6,174	6,174
	Log	5,633	5,565	5,564	5,564	5,564	5,564	5,564	5,564	5,564	5,564	5,564	5,564
	Gamma	2,079	1,970	1,967	1,963	1,968	1,963	1,963	1,963	1,963	1,963	1,963	1,963
Genetický algoritmus	LDRW	6,519	6,490	6,497	6,489	6,488	6,488	6,488	6,488	6,028	6,028	6,028	6,028
	Log	2,235	1,921	1,845	1,841	1,838	1,830	1,830	1,830	1,839	1,830	1,830	1,830
	Gamma	1,990	1,990	1,994	1,986	1,985	1,985	1,985	1,985	1,985	1,985	1,985	1,985
Simulované žíhání	LDRW	6,523	6,508	6,511	6,507	6,504	6,508	6,496	6,492	6,492	6,492	6,492	6,492
	Log	6,083	5,842	5,935	5,860	6,167	5,859	5,849	5,887	5,836	5,906	5,951	5,829
	Gamma	3,081	2,312	2,263	2,291	2,263	2,262	2,256	2,260	2,255	2,252	2,260	2,250
Střední funkční hodnoty	Iterace	100				1000							
	Populace	10	20	50	100	10	20	50	100	10	20	50	100
Netopýří algoritmus	LDRW	6,406	6,342	6,261	6,223	6,627	6,304	6,240	6,202	6,341	6,373	6,174	6,174
	Log	8,925	7,294	5,762	5,564	7,037	5,651	5,564	5,564	5,564	5,564	5,564	5,564
	Gamma	4,755	2,895	2,153	1,974	4,076	2,342	1,963	1,963	1,963	1,963	1,963	1,963
Genetický algoritmus	LDRW	6,718	6,527	6,515	6,499	6,554	6,488	6,488	6,488	6,029	6,028	6,028	6,028
	Log	6,452	2,824	1,968	1,950	5,025	1,846	1,830	1,830	2,930	1,851	1,830	1,830
	Gamma	2,336	2,056	2,034	2,005	2,241	1,985	1,985	1,985	2,007	1,985	1,985	1,985
Simulované žíhání	LDRW	7,038	6,851	6,619	6,645	6,665	6,653	6,590	6,557	6,516	6,507	6,505	6,513
	Log	9,172	7,892	7,087	6,506	7,866	7,021	6,185	6,745	7,175	6,614	6,380	6,140
	Gamma	6,106	3,680	2,749	2,442	2,569	3,062	2,662	2,637	2,378	2,313	2,291	2,320

B TABULKA VÝSLEDKŮ PRO DATA2

Tab. B.1: Tabulka hodnot pro data exp14_roimale_inp_tis_110907.mat

Minimální funkční hodnoty	Iterace	10			100			1000		
		10	20	50	100	10	20	50	100	1000
Netopýří algoritmus	Populace	10	20	50	100	10	20	50	100	1000
	LDRW	4,897	4,685	4,644	4,615	4,660	4,639	4,614	4,610	4,664
	Log	7,234	7,149	7,145	7,145	7,145	7,145	7,145	7,145	7,145
Genetický algoritmus	Gamma	14,168	13,775	13,748	13,722	13,725	13,721	13,721	13,721	13,721
	LDRW	4,664	4,686	4,645	4,648	4,645	4,638	4,638	4,638	4,641
	Log	6,902	6,879	6,890	6,885	6,917	6,875	6,875	6,875	6,880
Simulované žíhání	Gamma	16,546	16,488	16,478	16,464	16,514	16,463	16,463	16,463	16,463
	LDRW	4,972	4,703	4,672	4,709	4,684	4,705	4,710	4,652	4,663
	Log	10,232	9,855	9,993	9,789	10,074	10,295	9,747	9,773	9,790
Střední funkční hodnoty	Gamma	17,410	17,134	17,027	16,802	17,095	17,054	16,938	16,788	16,781
	Iterace	10	20	50	100	100	20	50	100	1000
	Populace	10	20	50	100	10	20	50	100	1000
Netopýří algoritmus	LDRW	9,445	6,942	4,954	4,766	7,363	5,690	4,803	4,684	6,503
	Log	12,707	9,216	7,473	7,148	11,874	7,505	7,145	7,145	7,145
	Gamma	17,735	19,412	14,459	14,004	17,290	15,300	13,802	13,721	14,675
Genetický algoritmus	LDRW	6,089	4,981	4,950	4,772	4,783	4,642	4,638	4,638	4,781
	Log	13,339	7,757	7,151	7,101	10,386	6,914	6,875	6,875	9,434
	Gamma	17,887	16,905	16,677	16,602	17,123	16,475	16,463	16,463	16,822
Simulované žíhání	LDRW	11,688	10,327	5,544	5,587	5,494	5,640	5,472	5,029	4,751
	Log	18,356	17,087	11,556	11,126	13,161	12,615	11,764	10,528	10,817
	Gamma	35,750	22,859	20,237	17,556	18,433	19,170	18,286	17,914	17,079

C TABULKA VÝSLEDKŮ PRO DATA3

Tab. C.1: Tabulka hodnot pro data exp17_roivelke_inp_tis_111007.mat

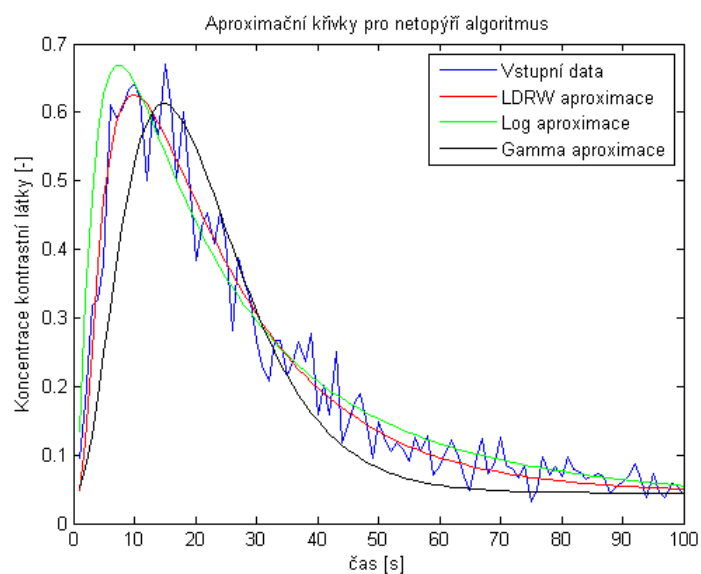
Minimální funkční hodnoty	Iterace	10			100			1000						
		Populace	10	20	50	100	10	20	50	100				
Netopýří algoritmus	LDRW		4,777	4,547	4,545	4,525	4,531	4,540	4,527	4,526	4,547	4,527	4,525	4,525
	Log		4,852	4,285	4,273	4,273	4,273	4,273	4,273	4,273	4,273	4,273	4,273	4,273
	Gamma		18,532	18,423	18,427	18,430	18,435	18,431	18,418	18,418	18,430	18,418	18,418	18,418
Genetický algoritmus	LDRW		3,771	3,789	3,810	3,737	3,737	3,737	3,737	3,737	3,740	3,737	3,737	3,737
	Log		6,192	6,150	6,166	6,153	6,156	6,145	6,145	6,145	6,154	6,145	6,145	6,145
	Gamma		21,971	21,946	21,966	21,947	21,944	21,935	21,935	21,935	21,936	21,935	21,935	21,935
Simulované žíhání	LDRW		3,968	3,829	3,809	3,806	3,760	3,918	3,795	3,810	3,741	3,745	3,742	3,740
	Log		6,290	5,991	6,028	6,006	6,005	5,982	5,999	6,037	6,110	6,010	5,980	5,999
	Gamma		26,641	25,411	25,346	25,432	25,231	25,913	25,424	25,355	25,214	25,225	25,234	25,221
Střední funkční hodnoty	Iterace		10			100	100			1000				
	Populace		10	20	50	100	10	20	50	100	10	20	50	100
Netopýří algoritmus	LDRW		6,497	5,858	4,790	4,619	6,418	5,153	4,696	4,565	5,420	4,674	4,593	4,526
	Log		7,502	6,154	4,323	4,273	5,391	4,273	4,273	4,273	4,273	4,273	4,273	4,273
	Gamma		25,763	21,108	19,000	18,580	25,801	18,968	18,783	18,441	19,247	18,470	18,418	18,418
Genetický algoritmus	LDRW		4,312	4,332	4,132	3,903	3,959	3,750	3,737	3,737	4,054	3,789	3,737	3,737
	Log		12,842	6,722	6,344	6,320	7,076	6,150	6,145	6,145	7,066	6,145	6,145	6,145
	Gamma		23,094	22,537	22,075	22,014	22,724	21,941	21,935	21,935	22,068	21,936	21,935	21,935
Simulované žíhání	LDRW		7,748	5,315	4,224	4,361	4,273	4,595	4,013	4,066	3,919	3,843	3,799	3,805
	Log		8,843	8,392	6,834	6,472	8,947	7,553	6,858	6,325	7,050	6,791	6,357	6,141
	Gamma		38,686	33,869	27,546	26,628	26,743	28,107	26,989	26,549	25,450	25,490	25,538	25,556

D TABULKA VÝSLEDKŮ PRO DATA4

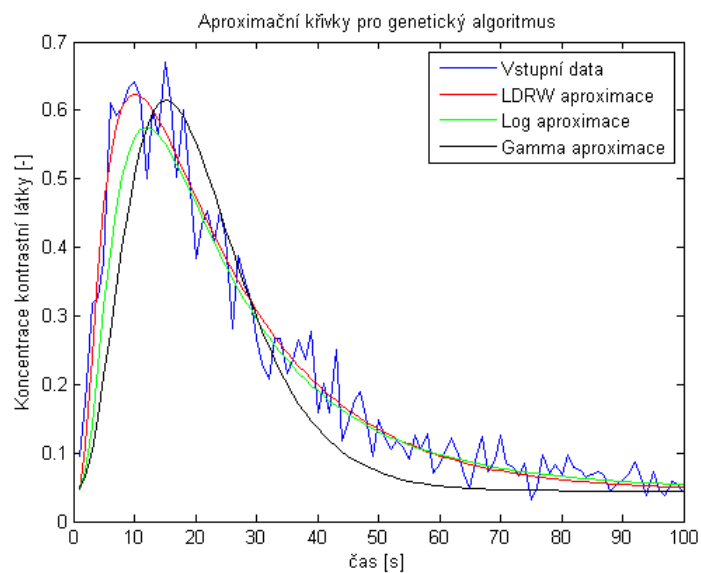
Tab. D.1: Tabulka hodnot pro data exp15_AIF_inp_tis_111007.mat

Minimální funkční hodnoty	Iterace	100					1000				
		10	20	50	100	100	10	20	50	100	100
Netopýří algoritmus	LDRW	9,335	9,328	9,319	9,320	9,357	9,325	9,315	9,316	9,324	9,315
	Log	7,167	7,156	7,156	7,155	7,155	7,155	7,155	7,155	7,155	7,155
	Gamma	4,714	4,276	4,262	4,262	4,329	4,261	4,261	4,261	4,261	4,261
Genetický algoritmus	LDRW	9,698	9,686	9,691	9,685	9,684	9,684	9,685	9,686	9,682	9,683
	Log	3,142	3,186	3,341	3,152	3,236	3,250	3,114	3,243	3,105	3,106
	Gamma	4,412	4,412	4,409	4,402	4,372	4,371	4,373	4,371	4,370	4,371
Simulované žíhání	LDRW	9,729	9,704	9,689	9,684	9,685	9,710	9,709	9,692	9,684	9,687
	Log	3,963	3,905	3,804	3,787	3,895	3,777	3,730	3,740	3,842	3,746
	Gamma	4,816	4,595	4,626	4,591	5,284	4,752	4,738	4,638	4,598	4,576
Střední funkční hodnoty	Iterace	100					1000				
		10	20	50	100	100	10	20	50	100	100
Netopýří algoritmus	LDRW	9,380	9,372	9,362	9,345	9,396	9,379	9,350	9,344	9,376	9,331
	Log	7,849	7,604	7,324	7,156	7,485	7,355	7,155	7,155	7,155	7,155
	Gamma	5,984	5,770	4,594	4,336	6,362	4,937	4,262	4,261	4,261	4,261
Genetický algoritmus	LDRW	9,747	9,741	9,722	9,712	9,698	9,700	9,697	9,702	9,682	9,687
	Log	6,154	5,363	4,270	3,623	5,820	4,257	3,458	3,561	3,253	3,189
	Gamma	5,985	5,240	4,832	4,575	4,475	4,395	4,393	4,402	4,370	4,374
Simulované žíhání	LDRW	9,903	9,770	9,721	9,733	9,738	9,741	9,743	9,728	9,693	9,701
	Log	6,637	5,158	4,445	4,181	6,394	4,936	4,165	4,035	4,441	4,083
	Gamma	8,455	7,140	6,314	5,321	6,244	5,564	5,538	5,673	4,799	4,755

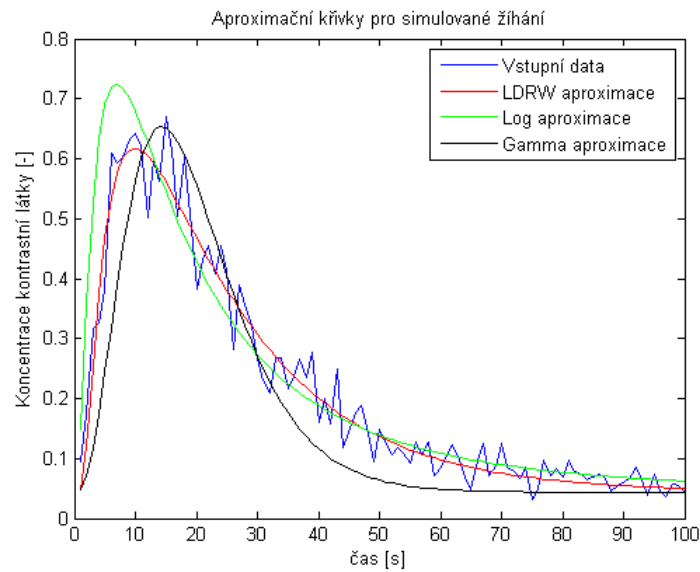
E GRAFY VÝSLEDKŮ PRO DATA2



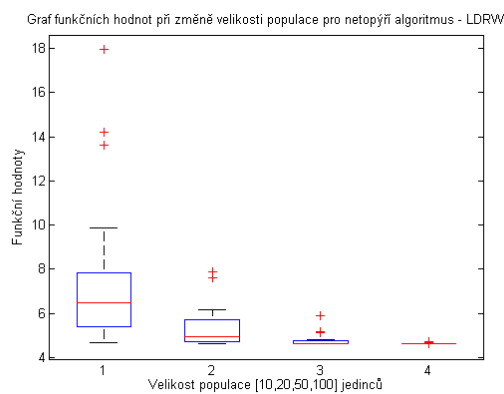
Obr. E.1: Aproximace dat pomocí netopýřího algoritmu



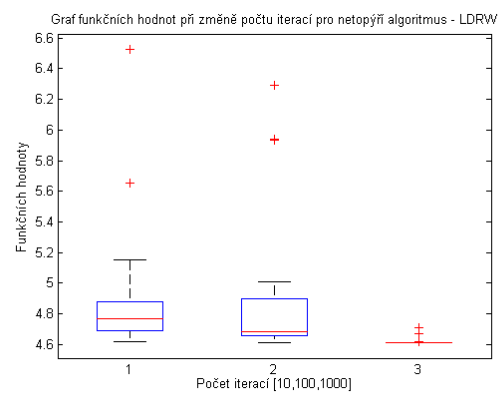
Obr. E.2: Aproximace dat pomocí genetického algoritmu



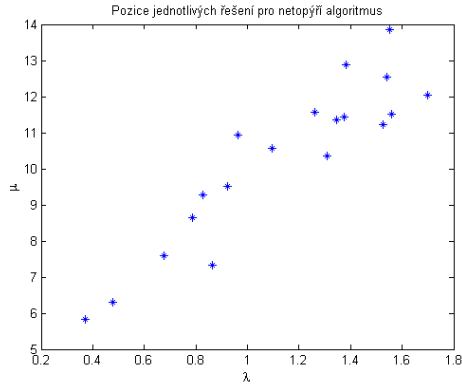
Obr. E.3: Aproximace dat pomocí simulovaného žihání



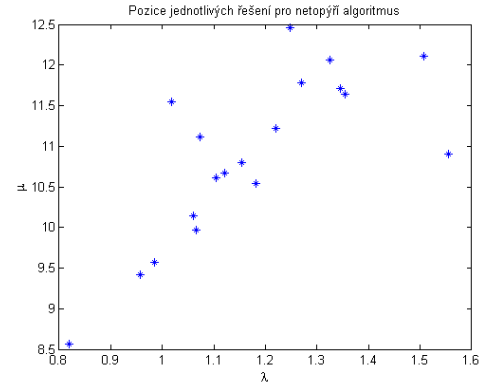
Obr. E.4: Srovnání výstupů pro změnu populace



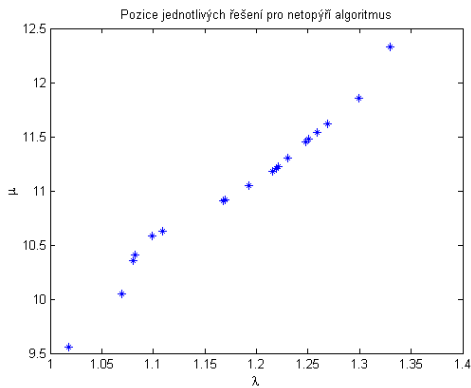
Obr. E.5: Srovnání výstupů pro změnu iterace



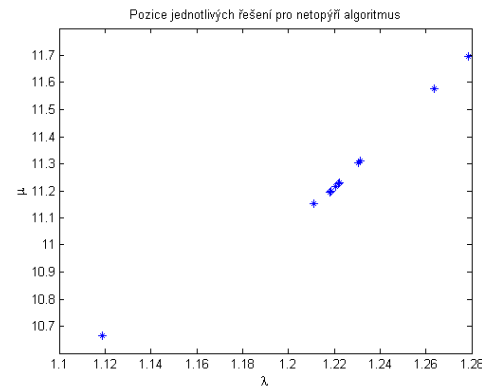
Obr. E.6: Pozice parametrů pro netopyř algoritmus, populace n=10



Obr. E.7: Pozice parametrů pro netopyř algoritmus, populace n=20



Obr. E.8: Pozice parametrů pro netopyř algoritmus, populace n=50

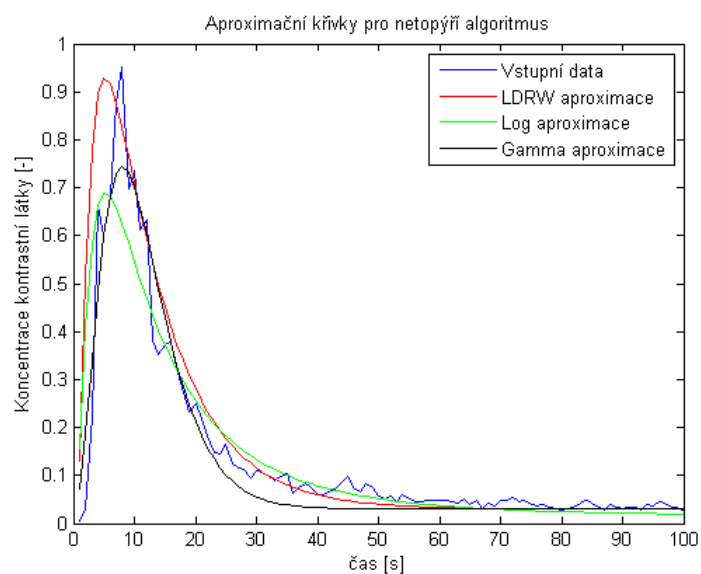


Obr. E.9: Pozice parametrů pro netopyř algoritmus, populace n=100

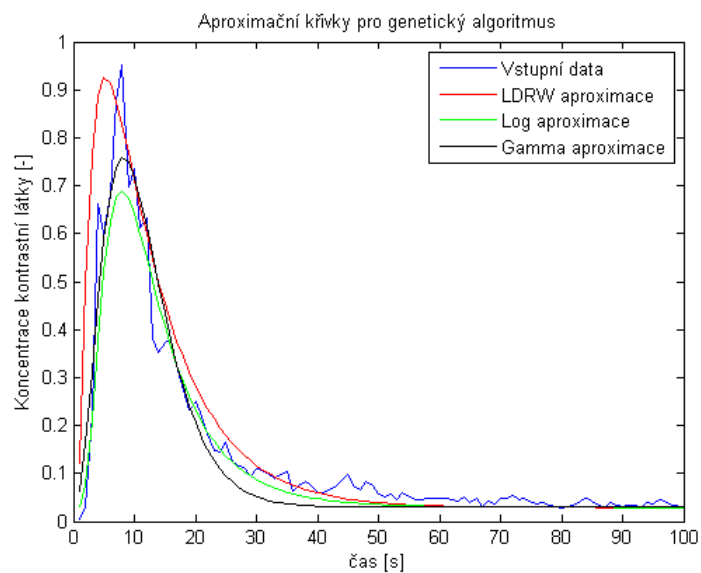
Grafy E.1, E.2, E.3 zobrazují vstupní křivku a aproximované křivky pomocí jednotlivých modelů a funkcí. Pro pomalá data se nejvíce hodí LDRW funkce. Na další straně je zobrazen graf E.4, na kterém je vidět rozdíl při změně velikosti populace. Na druhém grafu E.5 nejsou výsledky tak zřetelné. Tyto grafy jsou vytvořeny pro netopyř algoritmus. Tento graf je vytvořen pro velikost populace n=20. I přes větší počet iterací, se pro malý počet jedinců nedosáhlo lepšího výsledku. Proto je vhodnější použít počet iterací alespoň 1000 pro dané nastavení.

Stejné výsledky poskytují i další grafy E.6, E.7, E.8 a E.9. Na těchto grafech je vidět rozdíl pro změnu velikosti populace. Pokud je postupně zvyšována velikost populace, dochází také ke shlukování výsledných bodů.

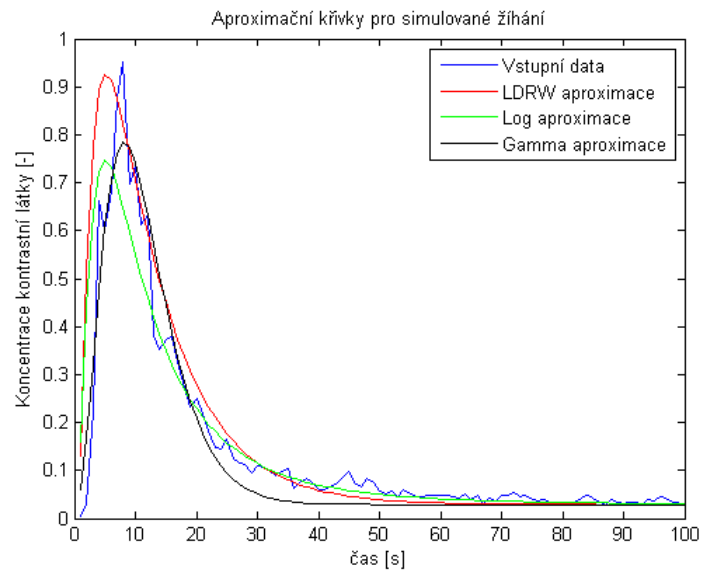
F GRAFY VÝSLEDKŮ PRO DATA4



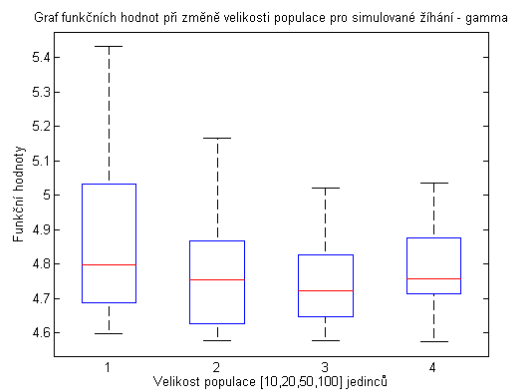
Obr. F.1: Aproximace dat pomocí netopýřího algoritmu



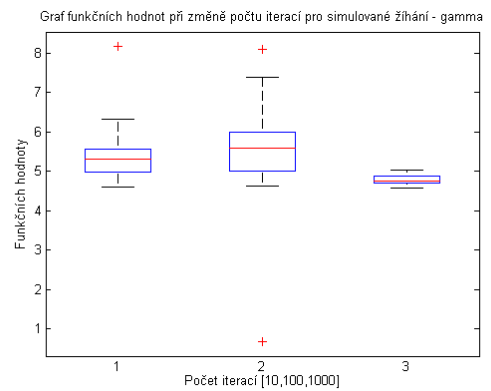
Obr. F.2: Aproximace dat pomocí genetického algoritmu



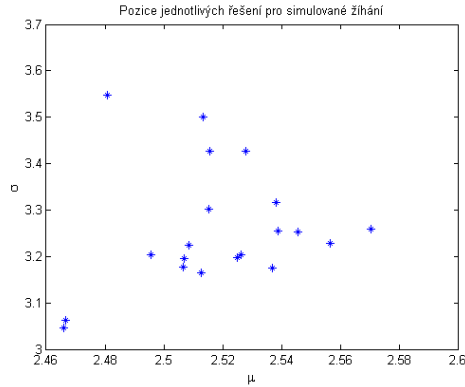
Obr. F.3: Aproximace dat pomocí simulovaného žihání



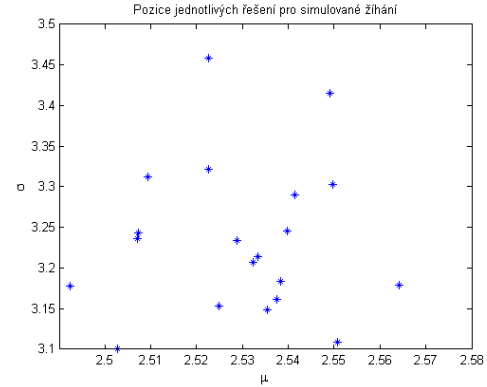
Obr. F.4: Srovnání výstupů pro změnu populace



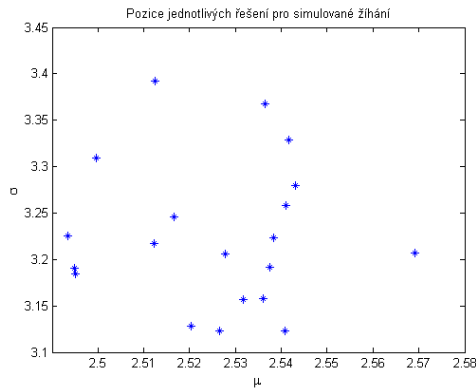
Obr. F.5: Srovnání výstupů pro změnu iterace



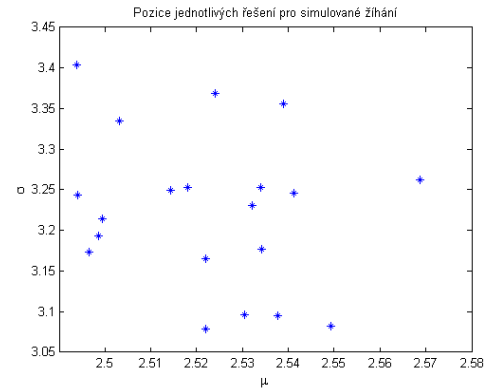
Obr. F.6: Pozice parametrů pro SA, populace $n=10$



Obr. F.7: Pozice parametrů pro SA, populace $n=20$



Obr. F.8: Pozice parametrů pro SA, populace $n=50$



Obr. F.9: Pozice parametrů pro SA, populace $n=100$

Pro data4, která spadají do kategorie dat rychlých, se jako nejlepší aproximační funkce jeví gamma funkce.

Protože tyto grafy jsou vytvořeny z algoritmu simulovaného žíhání, nejsou výsledky tak přesvědčivé jako u předchozích metod. Na grafu F.4 je vidět postupné zmenšení rozptylu, ovšem díky náhodnosti celého výpočetního procesu není i při maximální populaci nejlepší výsledek. Na grafu F.5 je zobrazena změna počtu iterací pro funkci gamma. Pro počet iterací 10 a 100 nejsou výsledky přesvědčivé. Pro počet iterací 1000 je již rozptyl hodnot velmi malý a dá se předpokládat že program dojde ke správnému výsledku. Pokud bychom chtěli stejné výsledky jakých dosahují programy pro netopýří a genetický algoritmus, bylo by nutné použít větší počet iterací.

G DRUHÁ PŘÍLOHA

Tab. G.1: Použité testovací nastavení

Nastavení	
Velikost populace	[10,20,50,100]
Opakování	[20]
Vzorkování	[1]
Počet iterací	[10,100,1000]
Amin, Amax	[0,100]
Bmin, Bmax	[0,100]
Koeficient mutace	[0.2]
k_net_ldrw	[0.35]
k_net_log	[1.3]
k_net_gamma	[0.5]
k_gen_ldrw	[0.35]
k_gen_log	[1]
k_gen_gamma	[0.5]
k_sa_ldrw	[0.35]
k_sa_log	[1.3]
k_sa_gamma	[0.5]
Počáteční teplota	[50]

Tab. G.2: Technické parametry testovacího počítače

Verze systému	Windows 7 Professional
Procesor	AMD Phenom™ X4 965, 3,40GHz
RAM	4,00GB
Grafická karta	NVIDIA GeForce GTX 460

H OBSAH PŘÍLOŽENÉHO CD

Tab. H.1: Seznam přiložených souborů na CD

Složka	Název souboru	Popis
Data		Jednotlivá vstupní data
Programy	GUI_v2.m gen_data_gamma.m gen_data_LDRW.m gen_data_log.m gen_spoj_gamma.m gen_spoj_LDRW.m gen_spoj_log.m netopyri_data_gamma.m netopyri_data_LDRW.m netopyri_data_log.m netopyri_gamma.m netopyri_LDRW.m netopyri_log.m SA_data_gamma.m SA_data_LDRW.m SA_data_log.m SA_gamma.m SA_LDRW.m SA_log.m mutace.m reprodukce.m	Uživatelské rozhraní Funkce pro skutečná data Funkce pro skutečná data Funkce pro skutečná data Funkce pro modelová data Funkce pro modelová data Funkce pro modelová data Funkce pro skutečná data Funkce pro skutečná data Funkce pro skutečná data Funkce pro modelová data Funkce pro modelová data Funkce pro modelová data Funkce pro skutečná data Funkce pro skutečná data Funkce pro skutečná data Funkce pro modelová data Funkce pro modelová data Funkce pro modelová data Funkce provádějící operaci mutace Funkce provádějící operaci reprodukce
Výstupy	exp13_AIF_inp_tis_111007.xlsx exp14_roimale_inp_tis_110907.xlsx exp15_AIF_inp_tis_111007_data4.xlsx exp17_roivelke_inp_tis_111007.xlsx	Výstupní tabulky a data Výstupní tabulky a data Výstupní tabulky a data Výstupní tabulky a data
Programy_SP		Programy pro modelování křivek, semestrální projekt
	Kriz_Marek_DP.pdf	Diplomová práce